UNIVERSITY OF CALIFORNIA
Los Angeles

Knowledge discovery from biomedical and scientific text

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Computer Science

by

Justin Wood

2022

ABSTRACT OF THE DISSERTATION

Knowledge discovery from biomedical and scientific text

by

Justin Wood
Doctor of Philosophy in Computer Science
University of California, Los Angeles, 2022
Professor Wei Wang, Chair

The world is overflowing with text. This ever-growing resource has the ability to capture thoughts, ideas, and understandings. One example is the scientific research paper which often contains a new discovery or details an in depth understanding—and new research papers are growing at an exponential rate, a rate that may be difficult for the human mind to keep up with. Additionally, the increase of collaboration across different knowledge domains requires an increased effort for a specialist of any one field to understand. These increases can lead to mistakes among researchers from missing important information that has already been documented. Similarly, the electronic health record is increasing rapidly as technology is integrating itself into the patient physician interaction. To be able to deliver information quickly and accurately to a physician can help ease the burden and lesson the mistakes that a primary care physician can make when dealing with the increasing pressure from seeing too many patients in too little time.

Given the enormous amount of textual data, computational techniques must be developed that can effectively process the data. This work presents approaches that seek knowledge discovery from a large input of biomedical and scientific text. In the context of scientific research papers, we discuss how and why we need to automate the scientific method using a causal pipeline. Starting with the raw text of a scientific corpus we demonstrate the ability to improve scientific decision making in experiment planning and deductions. For the task of summarizing corpora, we introduce a new topic model that seeks to model topics off a pre-existing knowledge source. As we show empirically, our methods of extraction, connection, and summarization of relevant electronic text records results in knowledge discovery and new understandings.

The dissertation of Justin Wood is approved.

Yizhou Sun

Carlo Zaniolo

Alcino Jose Silva

Corey Wells Arnold

Wei Wang, Committee Chair

University of California, Los Angeles

2022

We've developed a language that allows computers to reason like us. This will allow computers to assist those who are constantly on the lookout for cause and effect, such as researchers in the healthcare sciences, as well as in the social sciences. They care about improving drugs, they care about whether a drug is going to prevent a disease or harm your liver. They are concerned with narrowing economic inequality and slowing global warming. But until very recently, they didn't have a language in which to express their scientific knowledge and translate it into causal questions and conclusions. Once you are able to express your scientific knowledge mathematically, you can combine it with data and come up with conclusions that answer your questions.

—JUDEA PEARL

And the other idea, which also has been around, is causality. The idea that forming a prediction about the future is different from making a causal claim about the way the world works. A lot of the successes for AI and machine learning are now predictive in nature. They are taking the world as it comes to us and they are forming a prediction about the future and then using that prediction to succeed. But if we want to take the next step, and we want to know what would happen if I intervene in a certain way and how will that change the world, that's a causal problem.

—DAVID BLEI

Humans and machine algorithms have complementary strengths and weaknesses. Each uses different sources of information and strategies to make predictions and decisions...humans can improve the predictions of AI even when human accuracy is somewhat below [that of] the AI – and vice versa. And this accuracy is higher than combining predictions from two individuals or two AI algorithms.

—MARK STEYVERS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

During my time at UCLA I was privileged to be most heavily guided by three great professors: Corey Arnold, Alcino Silva, and Wei Wang. From Corey I learned to love topic modeling, to think in a Bayesian way. I am also grateful to Corey for adopting me into the medical imaging informatics lab, a place that provided me with tremendous people resources and context for interesting research. Alcino lent Nick and I his vision of revolutionizing the scientific method. It was in our weekly meetings where Alcino's passion fueled our delightful philosophical talks about ResearchMaps. Lastly, I owe a great deal of gratitude to my advisor, Wei. I have met few people in my life that have been as supportive as Wei. For any direction I wished to pursue she was always there to lend me advice or provide me with the resources needed to accomplish my goals. I will always remember the great deal of effort Wei gave to aiding my research, especially during paper deadlines, helping me with presentations and all those interesting stories.

I owe a special mention to my remaining committee members: Professor Carlo Zaniolo and Professor Yizhou Sun. Carlo's kindness, teaching direction and encouragement helped me a great deal throughout my graduate work; it is Carlo's passion and dedication to his field that serves as an inspiration not only for this dissertation, but for my post-graduate career as well. To Yizhou I am grateful for the all the last minute help, without it this dissertation would not be possible. I am also thankful for the teaching assistant opportunity along with the knowledge she gave to me during that time. It is from Yizhou's explanation of neural networks that provided the foundation for my topic modeling and recurrent neural network paper.

Professor Junghoo (John) Cho is an incredible influence and inspiration. From John I learned most of the information retrieval context found in this dissertation. Alongside his expertise, John provided me with such a joyful environment for learning. I was always looking forward to attending his lectures, as John would take complex topics and explain them in such amazing detail that they became easily absorbed.

Along the way I have been aided by a plethora of other wonderful professors. Professor William Hsu, Nick and I often teamed up on our many piecemeal causality papers. Will also gave me some great advice in how to conduct experiments for our "mind" paper. Professor Alex Bui has always been incredibly helpful, especially during my time in the biomedical big data training program. I enjoyed my talks with Professor Frank Meng, as they were always encouraging. We had a wonderful collaboration with Professor Peipei Ping and I am grateful for the opportunities she provided me to collaborate and network with many talented researchers. To Professor Alexej Abyzov I am thankful for the opportunity to learn about research at the Mayo Clinic and for providing me with an in-depth understanding of sequence alignment algorithms. And to professors: Steven Daniels, Chandra Chekuri, and Nancy Van Cleave, I am appreciative for all the help provided to me to get me started on my journey here at UCLA, whom with no incentive other than to ensure my success, went above and beyond what was asked of them.

My main collaborator is Nicholas Matiasz. It is in Nick's philosophical expertise of piecemeal causality together with incredible literary talent that much of our research was able to be successfully crafted. I have had few, if any, such rich collaborations. There have been of course a multitude of other students, staff, friends, and colleagues that helped along the way. I will try to name as many as I can, however there are many not shown. I would like to acknowledge Patrick Tan for an enjoyable collaboration on Aztec and for help on my paper, Source-LDA—and to Vincent Kyi, Brian Bleakley, Howard Choi, Noah Adler, Jiayun Li, Johnny Ho, Panayiotis Petousis, Karthik Sarma, Edgar A. Rios Piedra, Steve Arbuckle, Joseph Brown, Wenchao Yu, Ariyam Das, Jae Lee, Preeyada Imraruen, Artem Tartakynov, Devendren Reddy and Arul Jeyaraj—thank you all.

I would be amiss not to include some of the most influential people in my life. My father, James Wood II, instilled in me a love of mathematics and a curiosity for science. From my mother, Joyce Wood, I learned studiousness and how to succeed through hard work. During these times I have had the opportunity to spend more time with my family for which I am grateful. I have been incredibly fortunate to experience the joyous beginnings of my two nephews James Wood IV, Jackson Wood and niece Annabelle Wood. To my brother, James Wood III and sister-in-law Andrea, thank you so much for allowing me to share in these joyful moments. I am perhaps most grateful for the time in my life spent with my grandparents James Wood I and Lula Wood. To my grandmother, Lula Wood, I dedicate this dissertation.

Justin Wood
Los Angeles, CA
8 April 2022

VITA

2008	Bachelor of Science in Computer Science *cum laude*, University of Illinois
2008–2015	Software Engineer, Industry
2015-2022	Graduate Student Researcher in Computer Science, UCLA
2016	Research Intern, Mayo Clinic
2017-2021	Teaching Assistant in Computer Science, UCLA
2018	Master of Science in Computer Science, UCLA
2018–2020	Senior Software Engineer in Research, Agoda
2022 (expected)	Doctor of Philosophy in Computer Science, UCLA

PUBLICATIONS

Yu W, Das A, **Wood J**, Wang W, Zaniolo C, Luo P. Max-Intensity: Detecting Competitive Advertiser Communities in Sponsored Search Market. In: Proceedings of the IEEE International Conference on Data Mining (ICDM); 2015 Nov 14–17; Atlantic City, NJ. p. 569–578.

Matiasz NJ, **Wood J**, Hsu W, Silva AJ. ResearchMaps.org: A free web app for integrating and planning experiments. Poster session presented at: 15th Annual Molecular and Cellular Cognition Society Symposium; 2016 Nov 10–11; San Diego, CA.

Matiasz NJ, **Wood J**, Wang W, Silva AJ, Hsu W. Computer-aided experiment planning toward causal discovery in neuroscience. Frontiers in Neuroinformatics. 2017 Feb 13;11:Article 12.

Wood J, Tan P, Wang W, Arnold C. Source-LDA: Enhancing probabilistic topic models using prior knowledge sources. In: Proceedings of the IEEE 33rd International Conference on Data Engineering (ICDE); 2017 Apr 19–22; San Diego, CA. p. 411–422.

Matiasz NJ, **Wood J**, Wang W, Silva AJ, Hsu W. Translating literature into causal graphs: Toward automated experiment selection. In: Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM); 2017 Nov 13–16; Kansas City, MO. p. 573–576.

Matiasz NJ*, **Wood J***, Doshi P*, Speier W, Beckemeyer B, Wang W, Hsu W, Silva AJ. ResearchMaps.org for integrating and planning research. PLOS One. 2018 May 3;13:e0195271.

Wood J, Li B, Lee J, Arnold C, Wang W. On the Utility of Combining Topic Models and Recurrent Neural Networks. In: Proceedings of the Springer 17th International Conference on Computing and Information Technology (IC2IT); 2021 May 13–14; Bangkok, Thailand. p. 66–76.

Matiasz NJ*, **Wood J***, Wang W, Silva AJ, Hsu W. Experiment selection in meta-analytic piecemeal causal discovery. IEEE Access; 2021 Jul 1;9:97929-41.

Wood J, Wang W, Arnold C. The Biased Coin Flip Process for Nonparametric Topic Modeling. In: Proceedings of the Springer 16th International Conference on Document Analysis and Recognition (ICDAR); 2021 Sep 5–10; Lausanne, Switzerland. p. 68–83.

Wood J, Arnold C, Wang W. A Bayesian Topic Model for Human-Evaluated Interpretability. In: Proceedings of The 12th Language Resources and Evaluation Conference (LREC); 2022, Jun 20–25; Marseille, France. In press.

CHAPTER 1

Introduction

There exists an overabundance of text. So much so that it is impossible to manually curate all this data to make the same kinds of knowledge findings that a machine is capable of discovering. How can a machine do this? One way is to extract out all the relevant information from a given input set and then process the extracted data. A biological research paper, for example, consists of a complete understanding of an idea. But we may wish to only focus on the causality a research paper is conveying.

In the field of neuroscience (sub-field of biology), findings (which imply causality) are represented as outcomes observed from a target after intervening on an agent. It is in the research paper that neuroscientists document these findings. The collection of all neuroscience research papers represents the most complete database of causal assertions. What is needed is a way to store this data in a manner as succinctly as possible. A simplification of a causal assertion can thus be represented as a causal graph, where each node is an agent or target and the edge is the intervention and outcome. To store, construct, query and otherwise manage biological causal graphs, we develop a software application we call ResearchMaps.

With the framework in place for storing and representing causal assertions, we can then proceed to mine this data. One potential this data has is the ability to aid in experiment planning. In experiment planning, a major goal of the scientists is to plan experiments to uncover true causal assertions. Given the nature of causal understanding prior to any experiment planning as causal assertions that are connected together to form chains, these chains of causality can be represented as a partial graph with an arrow signifying a casual connection in the direction of the edge. Through careful experimentation, the research scientist is able to complete these graphs (every possible edge investigated) representing a casual network. But given a network representing only partial knowledge of true causality, how does the scientist determine which experiment to perform next? The answer is most doubtedly a qualitative one. We seek to shift this process to be a more quantitative one. By doing so, we have the potential to limit bias that hinders experiment planning and slows down the rate of scientific discovery. Additionally, by stitching together a network of causal assertions, called piecemeal causality, we show the ability to make meaningful deductions about the known causal world. These deductions include: a ranking of the most important associations, a score of how likely an association is to be discovered, and how likely an association is due to confounding variables.

ResearchMaps is an important first step in documenting the causal assertions of a research paper and modeling the causality in a way which is easy for data mining. The existing method of data input is by entering in the information by hand. While this most likely has the benefit of high precision, there is a lack of recall. The manual finding of the relevant information to input into ResearchMaps is an expensive one, even for the trained neuroscientist. It is thus unlikely given the existing method that the ResearchMaps database will grow sufficiently large enough to do intensive machine learning analysis. This lack of input data also hinders the current state of the art extraction methods, which are based off of machine learning and require a large amount of labeled input data. We are thus faced with the cyclic problem of needing data to find data. We seek to develop a new method for extracting causality using a small input size. By shifting away from traditional machine learning approaches we hope to leverage string alignment algorithms to find new causal assertions existing in text.

The extraction, representation, and data mining on the causal networks steps can be used to construction a pipeline starting with the research paper and ending with discoveries about the known causal world. We formalize this pipeline and demonstrate the results over an exhaustive collection of biological research papers. The results are presented, which we hope leads the biological community to new scientific discoveries. The pipeline represents a fundamental change to the scientific method, a shift from a qualitative approach to a quantitative one. As the research scientist is struggling to process the enormous amounts of textual data from an exponentially increasing corpus of research papers, so to is the physician with the electronic health record. The practice of medicine is undergoing a transformation driven by the rapid adoption of computational technologies. This transformation has been induced in part by technological developments that have enhanced the capability of health information technology (HIT) to streamline medical workflows, simplify billing practices, automate the collection of quality improvement (QI) data to improve outcomes, and much more. One important component of this transformation has been the widespread adoption of electronic health record (EHR) systems. These systems are intended to replace paper charts and allow for complete storage and retrieval of all medical records on a computer. The intent of these systems is to streamline workflows and increase efficiency by making it easier to index, find relevant information, and share it with members of a care team potentially in different geographic locations.

Increasing the usability of EHRs is one potential way to bring additional benefit to physician users. One important area of usability is better indexing and retrieval of a patient's clinical notes. Currently, it is often difficult for physicians to search through any one patient's clinical record and get a sense of what has been important to the patient over time. A system that could automatically detect what conditions have affected a patient at any given time by analyzing the text of the patient's medical record could allow for easy searching and visualization of a patient's history.

It is in the attempt to visualize the patient history that led to the development of Source-LDA [WTW17]. Topics provide a succinct overview of a given document; so by displaying the relevance of topics for a set of patient notes created over time, the physician can easily determine the important items needed to discuss or focus on when interacting with the patient. The application takes on the form of a multi-line chart with each line being the relevance of a topic to a patient's history. In order to facilitate the quick understanding of the topics and fit the topic descriptions in a single chart requires topic labels. Popular Bayesian techniques of topic modeling, which are usually extensions of latent Dirichlet allocation (LDA) [BNJ01], just return a distribution over the vocabulary of the corpus as the topic. It is unreasonable to ask the physician to decipher this distribution and map a label onto the topic mixtures. It must be the topic model itself or some post processing technique that assigns meaningful labels to the topics. Existing methods that label topics after inference (post processing) suffer from topics that contain a high assignment of words which cannot be easily assigned to a single topic. It is therefore more advantageous to integrate the labeling into the inference. This approach led us to the development of Source-LDA, a technique that represents the state-of-the-art technique for the labeling of topics and the related concept of topic interpretability. The key component to topic labeling and interpretability of Source-LDA is it's presumption of an outside knowledge source used to generate the corpus. The outside knowledge source is interpreted as topics, assumed to be a superset of topics used in the generative model, and then a subset of the knowledge source topics is used to bias the discovered topics of a Bayesian topic model.

At it's inception, Source-LDA was the best model among competing models, and it is still a competitive model—even among the more recently developed class of neural topic models—but it is far from complete. One problem the model has is deciding which topics from the knowledge source to use in the topic model. Since the desire is to make the process of selecting which labeled topics are used in the generation of the corpus require as little manual intervention as possible, we only require that a superset of topics be input into the model. For example if we are labeling patient notes, then we can easily get the possible topics by looking at MedlinePlus[1] (a consumer friendly medical encyclopedia). Using MedlinePlus as input to discover topics in patient notes produces good results, but the good results come at a significant cost. The superset of topics that are used as input significantly slow down the model. One complaint of people using Source-LDA is the speed. What is needed is a method that can be used to eliminate topics before inference that are unlikely to make it to the output—a set of topics we refer to as discarded topics. For this desideratum we introduce our ranking method, KnowledgeRank, which utilizes PageRank [PBM99] to order the topics by likelihood of being discarded.

Even though Source-LDA does a good job of labeling interpretable topics, it is always in the research scientist's aim to seek improvement. Source-LDA established itself as the best Bayesian-based interpretable topic labeling model because of it's flexibility to allow for assignment of words to topics even when they do not exist in the weakly-supervised knowledge source article. However, when semantically related words should be clustered together in a topic but do not cooccur in the knowledge source article, Source-LDA is lacking in it's ability to bind the related words together (even though it does allow for it). For example, suppose we have a Wikipedia article on the topic "Baseball" and for some reason the word "glove" is missing from the write-up. We would then say "glove" should belong to the topic "Baseball" even though it is not found in the knowledge source article. Topic modeling has techniques to handle this, for example if the words are showing up in the same documents than it is likely the topic model will assign them to the same topic. But can we

[1] https://www.nlm.nih.gov/medlineplus/

do better? And if so how? We propose a method to improve upon the assignment of words that should belong to a topic by leveraging the recurrent neural network (RNN). This deep learning technique has shown an astounding ability to generate segments of output that resemble the input used to train the model. These generated segments are often colloquially referred to as “dreaming”. Because these “dreams” are often similar to the trained input set but do not belong to the input set, we can ask the RNN whether or not a word that should belong to a topic is a “dream” produced after falling asleep while reading the knowledge source that corresponds to that topic. This method we refer to as the RNN enhanced Source-LDA (ReSource-LDA).

Further improvement of our general approach can be achieved in the amalgamation with nonparametric topic modeling. A somewhat unreasonable assumption of finite topic models is that the number of topics are known before hand. Nonparametric topic modeling provides the theoretical framework to remove this assumption. Dropping the need to prespecify the number of topics allows this number to be derived from the input—potentially leading to a more accurate partitioning of topics. We demonstrate the similarities between bounded and nonparametric inference techniques in a new interpretation of the Dirichlet process (the latent stochastic process that underpins nonparametric topic models) we call the biased coin flip process. The unboundedness of nonparametric topic models leads to the potential of an infinite-sized knowledge source input. Even with our topic filtering technique (KnowledgeRank) we are limited to a knowledge source size in the order of 10^6. Additionally, instead of filtering input before inference, we may be able to consider all input by utilizing the infinite nonparametric framework. Indeed, we demonstrate this possibility in our combination of weakly-supervised topic models (a class in which Source-LDA belongs to) and nonparametric topic modeling. In this combination we discover an interesting, novel topic model—a self-contained nonparametric topic model for highly-interpretable and labeled topics.

CHAPTER 2

Background

2.1 Piecemeal causality

2.1.1 Causality in biological literature

Much of biology is modeled as pathways, or signal cascades. Statements in the literature that describe causal relations between phenomena thus help biologists to understand these causal chains. And gaps in the understanding of a causal chain—i.e., its missing links—motivate hypotheses that then direct future research.

If an article is known to address the relation between two entities—e.g., a biological *agent*, A, and a biological *target*, T, then sentences in the article that mention both A and T are likely to describe that relation—for instance, in a results section that describes empirical results of experiments, or in a discussion section that describes the experiments' implications for the field.

Beyond simply stating that two entities are causally related, a sentence can also state (1) the type of experiment that was performed on the two entities, and (2) the result of the experiment. One way to express this type of biological evidence is with the *research map* representation [SM15]: Each experiment is either an *intervention* (involving an experimental manipulation) or an *observation* (involving no manipulation); within these two classes, the change of the agent can be either positive or negative, yielding four experiment classes: *positive intervention* ($\uparrow$), *negative intervention* ($\downarrow$), *positive observation* ($\varnothing^{\uparrow}$), and *negative observation* ($\varnothing^{\downarrow}$). The results of these experiments are categorized as either *increase* ($+$), *decrease* ($-$), or *no-change* (0) to indicate how the target responded to a change in the agent. Each pairing of an experiment class and a result class provides evidence for one of three types of relations: *excitatory* $\big((\uparrow,+);(\varnothing^{\uparrow},+);(\varnothing^{\downarrow},-);(\downarrow,-)\big)$, *inhibitory* $\big((\uparrow,-);(\varnothing^{\uparrow},-);(\varnothing^{\downarrow},+);(\downarrow,+)\big)$, and *independent* (any experiment that results in no change in the target).

2.1.1.1 Biological representations

The need to synthesize the vast amounts of biological data are given in various publications [SM15, LS13, MWW17a, MWD18]. An argument for the adoption of biological representations of existing knowledge are that the expansive set is too large for humans to process effectively. To that end, various different approaches have been developed to aid the scientist in understanding and utilizing this vast amount of scientific data.

One way to represent biological data is in the form of a probabilistic model [Fri04]. Probabilistic models have been effectively applied to different biological concepts such as cellular networks. Under such models, analysis and discovery have been enhanced after inference of the constructed representations. Other representations include causal graphs or networks [MWW17a]. Such work provides the groundwork for what information to take from a scientific article and how graphs can be constructed in order to select future experiments.

Other graph like forms can be biological ontologies, such as the Neuroscience Information Framework (NIF) [GAA08]. NIF is a general approach to synthesize nueroscience findings. The work includes a wide variety of data points and can serve as a biological ontology. NIF is comprehensive but is lacking a smaller focus on causality which can lead to a poor results in causal analysis. Other ontological representations come from Exelixis [KP11]. This project aims to track and represent ontologies that evolve over time. The work emphasizes a novel query language that facilitates an ontological search for the evolution of biological concepts. Our work, while more focused on representing and facilitating the querying of causality is similar in it's approach to graphical represent biological elements. Gene Ontology, another ontological representation, is a tool that synthesizes biological information into an ontological form. The project is ambitious and attempts to create a genomic vocabulary covering gene and protein roles. This tools shows the power of such data aggregation approaches as the Gene Ontology has become quite influential.

Networks again provide the setting in a more varied group of applications that model biological elements. A technique utilizing network based representations of biological elements have foundation in providing context for memory understanding [CAS16]. This work demonstrates that memories are linked together via neural ensembles when occurring close in time. The paper underscores the power of synthesizes biological processes into graphical networks to better understand some scientific domain. ResearchMaps [MWD18] is the basis for our pipeline method on representing biological causality graphically. The tool is also the main source of data for causal text fragments. The goal of Researchmaps is to represent a research paper as a casual network allowing for a more succinct yet powerful representation of a research paper. When the graph is connected together we form an exciting environment for knowledge discovery. A third network based method in a more general domain is WatsonPaths [LBB17]. WatsonPaths establishes the utility of using graphical based models for knowledge discovery. Another general biological representation is shown via BioCarta [Nis01]. This application is an example of an existing science based application that has enabled scientists an easier navigation of science-based information. The application specifically synthesizes biological pathways into connected graphical pathways. This work shows how the stripped down nature of textual content can be beneficial for understanding of mass amounts of information.

2.1.2 Causal extraction

Much of the early work in causal extraction was performed using some form of predefined knowledge bases [CC04, KB91, Hua21, HY10] or rule-based techniques [Che21b, GM02, PB21]. One example of a knowledge base approach is advance by [KB91]. In this work, the authors developed a system that integrates different components for the task of causal extraction. These combined techniques help form a knowledge base of textual causal relationships. Other techniques take as input pre-existing knowledge bases such as Freebase [Hua21] or WordNet [Gir10]. The knowledge bases can then be used as input to an algorithm to extract causality. While the effectiveness has been established in their respective settings, these techniques based on a knowledge base may not be suitable for our case as our vocabulary is quite esoteric without much supervised data. Also the knowledge base is only applicable to a single language.

Rule based approaches [Che21b, GM02, PB21] are somewhat similar to knowledge based approaches in that they rely on a fixed and previous established input. For example, [Che21b] use a rule based system for extraction of causal statements pertaining to the stock market. The rule based system was formed using a supervised input set. This work may be difficult in our setting because the rule-based technique is static to the input set and the input set itself may be difficult to obtain. Another approach advanced by [PB21] looks at six different issues related to causality. The work uses a model based on natural-language to extract or acquire causal statements. The rule based nature of the work limits the scope to that of general causality and is not transferable across languages.

Recently, focus has moved to more deep learning based approaches [Cao21, Gu21, KAS, BS21, Zha21]. One such method, based on a convolutional neural network, is proposed by [Zha21] to isolate context over sentences. The technique utilizes distant sentences to weigh the the labeling using a convolutional neural network and conveys better results than baseline methods. Another convolutional approach models relations as a dependency tree and inputs them into a graph convolutional network [Cao21]. These examples achieve good results over a large training set but also highlights the limitations. The large amount of training data required may not be attainable in the causal biological world. The convolution neural network may not be the best model in our setting as we are limited in supervised input. Another large collection of works takes a learning approach based on entity embeddings [Gu21, KAS, BS21]. Word embeddings are the foundation for causal search in a set of learning models [BS21]. These models are demonstrated to be applicable across different languages. A technique which utilizes BERT based vector representations is given by the model SCIBERT. The approach utilizes unsupervised scientific text to build vector representations similar to BERT, which can be used in a similar manner [Gu21]. BERT is again used in a transfer learning approach that is shown the improve causal sentence detection in some datasets using a bidirection gru with self attention combined with ELMO and BERT [KAS]. Embeddings based approahces while applicable on some datasets, may not be well suited for a more complicated set of data where causality is explained in esoteric terms. The focus on supervised data necessitates the model to learning more general textual patterns.

To allow for knowledge discovery in a setting without much training data, zero-learning approaches attempt to extract causality or related tasks without being solely dependent on deep learning models [Tau20, Do11, Lyu21, GE21]. Such techniques can use dependency graphs to search scientific text by focusing on patterns obtained from a labeled input set. In [Tau20], the authors are able to use a dependency graph for extractive search while adding a new query language for searching scientific based linguistic patterns. Differing zero-shot approaches focuses on examining the similarity among causal sentences [Do11], transfer learning to extract events from textual data [Lyu21] or crowd-sourced input [GE21]. However, these approaches are defined in the general domain, and not specific to causality extraction from biological text.

2.1.2.1 Sequence Alignments

Sequence alignment algorithms were primarily applied in the context of comparing genetic sequences [NW70, HC03, AG11, SW81]. Starting with a dynamic programming algorithm to compare the entirety of two sequences [NW70], the works evolved to comparing only singular subsequences [SW81] to multiple subsequences [HC03, AG11]. The latter [AG11] being an algorithm that takes an input k, and proceeds to find the optimal k breaks in the two sequences that result in the maximal local alignment scores. More recent work has recognized the ability of sequence alignment algorithms to be useful outside the context of genetic sequences [Wad21, WMS19].

Sequence alignment algorithms seek to assign a score for an alignment between two strings (A and B). The two most popular algorithms are Smith-Waterman [SW81] (local) and Needleman-Wunsch [NW70] (global). A local alignment is a maximal scoring alignment over the subsequences $A_p, A_{p+1}, A_{p+2}, \ldots, A_q$ and $B_x, B_{x+1}, B_{x+2}, \ldots, B_y$. A global alignment is the maximal scoring alignment over the A and B. Aligned strings often contain one or more instances of an *insertion* (or, interchangeably, *deletion*), which represents a single-character gap in the alignments. For example, with the alignments of the strings “BAT” and “BEAM” a global alignment could easily be:

```
B  –  A  T
|     |  |
B  E  A  M
```

With the “E” representing a deletion in the string “BEAM” and an insertion in “BAT”. Also note that the “T” in “BAT” is aligned to the “M” in “BEAM”; since they are not the same, this is referred to as a *mismatch*. To find the alignments, most algorithms use dynamic programming with one or more two variable recurrence relations stored in a matrix. We demonstrate the matrix using the alignment of “BAT” and “BEAM” and a scoring of match/mismatch ± 1 and an indel (insertion/deletion) score of -2.

		B	E	A	M
	0	-1	-2	-3	-4
B	-1	↖1	←-1	-3	-4
A	-2	-1	0	↖0	-2
T	-3	-3	-2	-1	↖-1

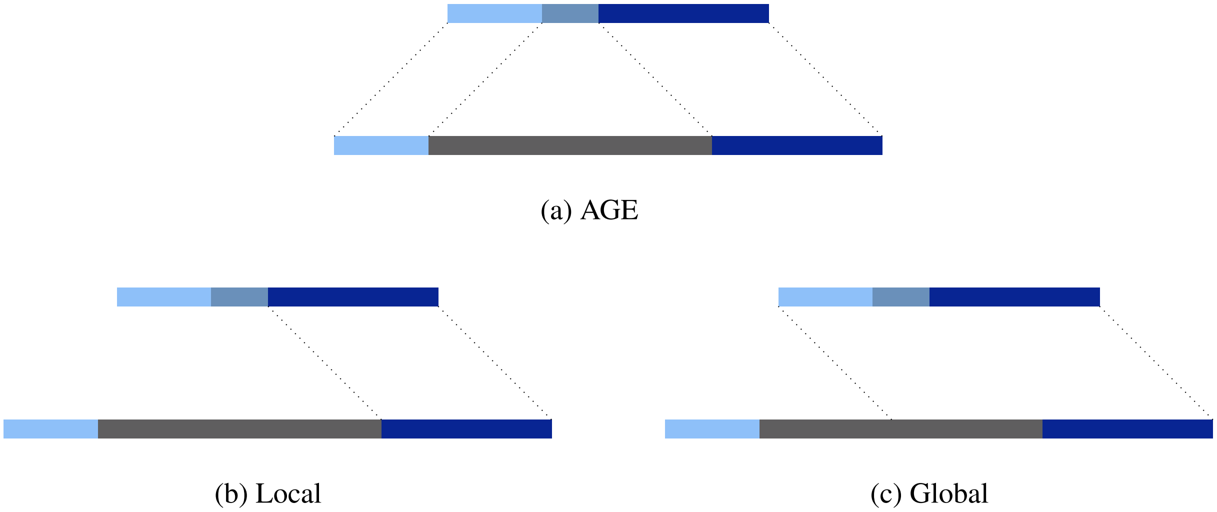
2.1.2.2 Smith-Waterman Algorithm

The Smith-Waterman algorithm [SW81] is a dynamic programming solution to the problem of finding an optimal local alignment. A local alignment over two sequences A and B is a maximal scoring alignment over the subsequences $A_p, A_{p+1}, A_{p+2}, \ldots, A_q$ and $B_x, B_{x+1}, B_{x+2}, \ldots, B_y$. An alignment with a linear gap is scored by assesing values for a match/mismatch S, and insertion/deletion Q. The solution to solving this maximal subsequence matching is given by the dynamic programming algorithm with recurrence:

$$M(i,j) = Max \left\{ \begin{array}{l} 0 \\ M(i-1,j-1) + S(A_i, B_j) \\ M(i-1,j) + Q \\ M(i,j-1) + Q \end{array} \right\} \tag{2.1}$$

This algorithm as shown can compute and return the maximal subsequence match in $\mathcal{O}(n^2)$ time and space. The linear gap however is not practical in such situations where it is more likely to see clusters of insertions/deletions than many non connected insertions/deletions.

For this reason it is common practice to use an affine gap penalty [Got82]. The change requires there to be an additional two recurrences. One for holding a transition to the insertion state (I) and the other for a transition into the deletion state (D). The M matrix then becomes the match/mismatch state. These new recurrences are given as:



(a) AGE

(b) Local

(c) Global

Figure 2.1: A visualization of two genomic sequences and their proper alignment which is formed by AGE (a). Both the global alignment (c) and local alignment(b) algorithms are unable to properly align the sequences to their optimal alignment which consists of two local alignments with a large gap in the bottom sequence.

$$M(i,j) = Max \left\{ \begin{array}{l} 0 \\ M(i-1,j-1) + S(A_i, B_j) \\ I(i-1,j-1) + S(A_i, B_j) \\ D(i-1,j-1) + S(A_i, B_j) \end{array} \right\} \tag{2.2}$$

$$I(i,j) = Max \left\{ \begin{array}{l} M(i-1,j) + O + Q \\ D(i-1,j) + O + Q \\ I(i-1,j) + Q \end{array} \right\} \tag{2.3}$$

$$D(i,j) = Max \left\{ \begin{array}{l} M(i,j-1) + O + Q \\ I(i,j-1) + O + Q \\ D(i,j-1) + Q \end{array} \right\} \tag{2.4}$$

2.1.2.3 AGE

Alignment with Gap Excision (AGE) [AG11] is an alignment algorithm that rectifies the problem of optimally aligning sequences that contain a large amount of insertions or deletions. It is shown that tuning the parameters of the Smith-Waterman [SW81] and Needleman-Wunsch [NW70] do not guarantee the optimal alignment equivalent to local alignments performed on both ends of certain sequences (shown visually in Figure 2.1). AGE is able to solve this problem by introducing a maximum matrix into the recurrence that hold the maximum value of the equivalent location in the local alignment matrix. This coupled with two local alignments, one going forward from the left end (L), and one going backward from the right end (R), guarantee a maximal split of local alignment scores. The algorithm guarantees finding the maximal right and left local alignments in both quadratic time and space. Even though the space is polynomial, AGE can be unusable with a large input size. To rectify this, a linear space version can be formulated as:

$$L(i,j) = Max \left\{ \begin{array}{l} 0 \\ L(i-1,j-1) + S(A_i, B_j) \\ L(i-1,j) + Q \\ L(i,j-1) + Q \end{array} \right\} \tag{2.5}$$

$$M(i,j) = Max \left\{ \begin{array}{l} L(i,j) \\ M(i-1,j) \\ M(i,j-1) \end{array} \right\} \tag{2.6}$$

$$R(i,j) = Max \left\{ \begin{array}{l} 0 \\ R(i-1,j-1) + S(a_i, b_j) \\ R(i-1,j) + Q \\ R(i,j-1) + Q \\ M(i-1,j-1) + S(a_i, b_j) \end{array} \right\} \tag{2.7}$$

$$D_b(i,j) = Max \left\{ \begin{array}{l} I_b(i,j+1) + O + E \\ G_b(i,j+1) + O + E \\ D_b(i,j+1) + E \end{array} \right\} \tag{2.8}$$

This algorithm can thus be used with linear space, requiring a memory bound of $\mathcal{O}(n)$ with computation time remaining at $\mathcal{O}(n^2)$.

2.1.2.4 NLP Search and Extraction

A similar objective to causality discover is that of scientific search. For knowledge discovery over scientific textual input, approaches utilize syntactic patterns [Shl20], patterns over dependency graphs [Tau20], and using preexisting machine learning based methods [Gu21, BS21, KAS]. Similarly, although not between scientific texts, search between parallel texts can be done effectively though word alignments [Wad21]. This analysis also showed the applicability to corpora across different languages.

Akin to the goals of causality extraction is that of relation extraction. The desiderata in relation extraction to discover relationships between different fragments in text. For example, one technique to causal discovery without labeled data is shown using a predefined database [Hua21] to connect text fragments. Other methods utilize graph-based techniques [Cao21], injected side information [GE21], or neural networks to achieve superior improvements over baseline models [Zha21].

A third extraction task, event extraction, is well-suited for zero-shot learning [Lyu21]. This allows extraction to be done in a way where annotation is not necessary—which is how event extraction is commonly done. In this setting transfer learning can be applied via neural networks to obtain results comparable to supervised methods.

2.1.3 Visualizing causality

As a representation for ontological information (i.e., entities and their relations), graphical causal models [SGS00, KF09, Pea09] can been used as a tool for experiment planning [Pea95]. Graphical models are a sensible formalism for guiding causal discovery: graphs concisely encode probabilistic relations between variables [Fri04]; they are accessible to domain experts because they encode plain causal statements (as opposed to only statistical or probabilistic ones) [Pea95, Pea09]; and principled methods exist for assembling fragments of graphical models into one [Fri04, Coh15], a strategy that resembles the way researchers integrate facts from various sources. An example graphical representation of causality is given by ResearchMaps. ResearchMaps, in addition to ontological information, includes epistemological (specifically, methodological) information regarding the evidence behind causal assertions.

2.1.3.1 Graphs of causality

A causal model can encode the causal structure of its variables with a causal graph. A causal graph is a directed graph with a set of variables (nodes) and a set of directed edges among the variables. A directed edge between two variables in the graph means that the variable at the tail of the edge has a direct causal effect on the variable at the head [SGS00, Pea09].

Via its structure (i.e., its connectivity), a causal graph encodes probabilistic dependence and independence relations. The graphical criterion known as d-separation [Pea09] can be used to read such relations of a causal graph; d-separation thus translates the edges of a graph into probabilistic statements. There is a key connection between d-separation and probabilistic independence relations: considering a directed acyclic graph (DAG) with the causal Markov and causal faithfulness assumptions [SGS00], any independence implied by d-separation holds if and only if the probability distribution associated with this DAG also exhibits this independence [Pea09].

2.1.3.2 Markov Equivalence Classes

Per the rules of d-separation, even if two or more causal graphs have different structures, they can encode the same (in)dependencies. A set of causal graphs that all imply the same (in)dependencies is called a Markov equivalence class [Pea09], or simply an equivalence class. An example of an equivalence class consisting of three unique graphs could be: $X \rightarrow Y \rightarrow Z$; $X \leftarrow Y \rightarrow Z$; and $X \leftarrow Y \leftarrow Z$. Although the graphs disagree on the orientation of the edges, they all imply the same (in)dependence relations: $X \not\perp\!\!\!\perp Y$; $Y \not\perp\!\!\!\perp Z$; $X \not\perp\!\!\!\perp Z$; and $X \perp\!\!\!\perp Z|Y$. Thus, these graphs are observationally Markov equivalent—i.e., they are indistinguishable given only the observed (in)dependence relations.

It is important to note that an equivalence class can be extremely large; the number of possible causal graphs is super-exponential in the number of variables in the model. For a system with only six variables, there are over three million possible causal graphs [Rob73]; if we allow for feedback (cyclicity), there are 230 million possible graphs. Causal discovery algorithms (that is, methods to identify the causal structure of a system) often cannot fully specify a single causal graph that accounts for the data; instead, they identify an equivalence class of graphs that satisfy the given (in)dependence relations. With only observational data, the graphs in an equivalence class will share the same adjacencies and vary in their edges' orientations. Interventional data, where the experimenter manipulates one of the variables, can eliminate specific causal structures from consideration.

2.1.3.3 Related representations

Existing representations such as Knowledge Engineering from Experimental Design (KEfED) provide a way to model experimental procedures and findings in a detailed and machine-readable manner [RRH11]. Other representations that represent formalisms such as probabilistic graphical models (e.g., Bayesian networks) have been shown to be effective at conveying relations among biological phenomena using a graph structure, as they compactly encode the joint probability distribution across variables [Fri04]. However, conditional probabilities are often missing in reports of experiments designed to test causal assertions. To display representations, pathway analysis tools such as BioCarta [Nis01] and Ingenuity Pathways Analysis (QIAGEN Redwood City, Redwood City, CA, USA) provide graphical interfaces of possible causal connections, but they do not keep track of the classes of experiments carried out to arrive at those conclusions, and they are usually restricted to specific domains of biological phenomena (e.g., molecular interactions).

2.1.4 Experiment selection

2.1.4.1 Causal discovery

Identifying the true causal graph for a system is the goal of a field known as *causal discovery* [Ebe17]. Each causal graph that can be drawn—with its unique structure—can be considered a particular explanation for the system it models. The goal of causal discovery is to find the one causal graph that correctly models the system—i.e., the correct explanation for the system's behavior. Knowing the true causal graph allows us to predict how the system will behave, including when we intervene on it.

Causal discovery is possible due to *bridge principles*, which "connect what can be observed to the underlying causal structure that generates the phenomena" [Ebe09]. The bridge principles we use here are two assumptions known as the causal Markov condition and the causal faithfulness condition. Together, these conditions allow for a relation between independencies in a probability distribution and edges in a causal graph [SGS00]. This relation thus allows us to infer features of a system's causal graph based on statistical relations that we derive from studies. For instance, if two variables in a system are statistically dependent, the causal graph that models the system will have certain features, such as one or more specific paths that correspond to this statistical dependence. The rules that these relations follow are given in the theory of *d-separation* [Pea09].

We express causal-structure constraints in the form $X \perp\!\!\!\perp Y \mid \mathbf{C} \mid\mid \mathbf{J}$. In this notation, X and Y are two variables that are statistically independent. This independence may have been inferred by statistically conditioning; the set $\mathbf{C}$ indicates the variables on which we conditioned to infer the independence. Similarly, the independence may have been inferred from an experiment in which one or more variables were intervened on; the set $\mathbf{J}$ indicates the variables that underwent experimental intervention when the relation manifested [HEJ14]. Both $\mathbf{C}$ and $\mathbf{J}$ can be the empty set ($\varnothing$). Dependence statements have the same form but instead use the "*not*-independent" symbol ($\not\perp\!\!\!\perp$). For example, the dependence relation

$$\text{long-term potentiation} \not\perp\!\!\!\perp \text{spatial learning} \mid \varnothing \mid\mid \text{long-term potentiation},$$

conveys that long-term potentiation and spatial learning were observed to be correlated in an experiment that intervened on long-term potentiation; in this case, no variables were statistically conditioned on to infer this independence.

The inference from statistical relations to causal graphs is not trivial: a set of (in)dependence relations may imply not just one graph but an equivalence class—a set of multiple graphs that are all equally consistent with the relations. An example of an equivalence class is these three causal graphs:

- $X \to Y \to Z$,
- $X \leftarrow Y \to Z$,
- $X \leftarrow Y \leftarrow Z$.

Each of these causal graphs is equally consistent with the following statistical relations:

- $X \not\perp\!\!\!\perp Y$,

- $Y \not\perp\!\!\!\perp Z$,
- $Z \not\perp\!\!\!\perp X$,
- $X \perp\!\!\!\perp Z \mid Y$.

Given a set of (in)dependence relations over a set of variables, it is not immediately obvious which causal graphs are consistent with the relations. In principle, a researcher could derive the equivalence class by hand; however, this manual computation is infeasible for all but the simplest of cases. And causal inference is further complicated by conflicting information. For instance, one experiment may suggest that two variables are dependent, while another experiment may suggest that they are independent. A principled approach to causal discovery should include a method to resolve such conflicts.

2.1.4.2 Constraint-based causal discovery

The strategy known as *constraint-based causal discovery* is to express information about a system in the form of logical propositions, which serve as constraints on causal structure. These constraints then guide how an algorithm searches for the set of causal graphs that are optimal, according to some optimization criterion. Although research articles are often unaccompanied by the primary data that underlie them, articles often contain constraints implicitly in the form of statistical relations, statements indicating either a dependence or independence between two variables. Beyond stating that variables are (in)dependent, an (in)dependence relation may be qualified by additional context: the relation may have been observed only when one or more other variables were statistically conditioned on, or when one or more variables were intervened on—or both.

One particular constraint-based algorithm—developed by [HEJ14]—represents the current state-of-the-art algorithm in causal discovery. Among current methods, it considers the most general model space: neither acyclicity nor causal sufficiency needs to be assumed—the algorithm can thus consider models that contain both cycles (feedback) and latent confounders. Additionally, the algorithm's constraint-based approach enables the formalization of background assumptions [Ebe17], as well as the degree-of-freedom approach described in Section 3.4.1.1.

The intuition for this constraint-based algorithm is as follows: Scientists will perform experiments to understand the causal relations that govern the phenomena in a system. These phenomena and the causal relations between them can be represented by the nodes and directed edges that compose a causal graph. We call this causal graph that correctly models the system the *true* causal graph. In addition to this true graph, there are other graphs with the same variables but different sets of edges, corresponding to different causal descriptions of the system's phenomena. The number of possible causal graphs is large, even for small sets of variables. Thus, the scientist who performs experiments to identify the true causal graph is "searching for a needle in a really huge haystack of falsehoods" [Gly04].

An experiment's result can show the scientists which parts of the haystack are safe to remove: namely, all the causal graphs that are inconsistent with the result.[1] When a result is expressed as an (in)dependence relation, the rules of d-separation can be used to identify the particular causal graphs that are consistent with the result. Any scientist who understands d-separation can use a pen and paper to check whether an (in)dependence relation is consistent with a causal graph. But this computation is infeasible to do manually when there are thousands of possible graphs, as is true even for a system with only five variables. Therefore, the strategy taken by [HEJ14] is to have this done computationally.

The algorithm uses answer set programming (ASP), a type of logic programming that is useful for solving very challenging problems such as NP-hard optimization tasks. It is based on the concept of declarative constraint satisfaction [GL88, Bar03]. In this context, the constraints are (in)dependence relations, and they are satisfied only by the particular causal graphs that encode those relations, as given by the rules for d-separation.

The algorithm proceeds in the following steps: First, (in)dependence relations among the system's variables are obtained—either by performing statistical independence tests on data [HEJ14], or by annotating statistical relations that are reported in the literature, as is done with the *ResearchMaps* web application [MWW17a, MWW17b, MWD18]. If none of the constraints conflict with each other, then a Boolean satisfiability (SAT) solver [BHM09] is sufficient to find the consistent causal

[1] Of course, an erroneous result can mislead the scientists by motivating them to remove a part of the haystack that in fact contains the needle (i.e., the true causal graph). The model discounts the scientist's fallibility; instead, the focus is on how to reason with evidence and plan experiments, assuming that those experiments will be performed competently.

graphs [HHE13]. However, if the constraints contain conflicts—for instance, if one constraint states that X and Y are independent, while another states that they are dependent—then a Boolean *maximum* satisfiability (MaxSAT) solver is required. In this case, each constraint is assigned a weight that denotes its confidence, and the solver finds the causal graphs that minimize the sum of the weights for unsatisfied constraints [BHM09]. Weights can be assigned based on the p-values of independence tests [HEJ14] or based on other measures of confidence, such as the evidence score for the research-map edge from which the constraint was derived (see Section 3.3.3). [HEJ14] formulate the search for (maximally) consistent causal graphs as a constrained optimization problem. For the (in)dependence constraints $\mathbf{K}$ over the variables $\mathbf{V}$, each with a non-negative weight $w(k)$, we search through a class of graphs, $\mathcal{G}$, to find the causal graph G^* such that

$$G^* \in \operatorname*{argmin}_{G \in \mathcal{G}} \sum_{k \in \mathbf{K} : G \not\models k} w(k)\,, \tag{2.9}$$

where $G \not\models k$ states that the causal structure of G does *not* imply the constraint k. We thus wish to find the causal graphs that minimize the summed weight of unsatisfied constraints. A state-of-the-art MaxSAT solver named Clingo [GKK11] is guaranteed to converge to a globally optimal solution, thus identifying the one or more causal graphs that maximally satisfy the constraints.

To accommodate both conflicting and conflict-free sets of evidence, here we use the phrase “equivalence class” in two ways: (1) to refer to a Markov equivalence class, as traditionally defined [SGS00]; and (2) to refer to the set of causal graphs that satisfy Equation 2.9. This second meaning addresses the fact that conflicts can be resolved in multiple ways. Depending on how the conflict is resolved—and which evidence is discarded to achieve this resolution—different sets of graphs will be considered consistent. In this case, “equivalence class” denotes the set of causal graphs that remain consistent with the evidence that one is currently willing to consider. Unless otherwise specified, we intend this second meaning throughout this dissertation.

2.1.4.3 Degrees of freedom

An equivalence class of causal graphs represents the range of causal interpretations one can defensibly take in light of the available evidence. The diversity of causal structures in an equivalence class represents the extent to which the available evidence is lacking and the extent to which the true causal graph is *underdetermined*: the less evidence there is, the more causal graphs will remain that are consistent with what is known. Because this lack of knowledge is what drives scientific inquiry, quantifying a causal graph's underdetermination can help scientists to determine which next experiments could be most instructive. We can quantify this underdetermination by considering the diversity of causal structures that exist throughout all the graphs in an equivalence class.

The *degrees of freedom* for a causal graph are the possible variations in edge relations that can exist between any two variables throughout an equivalence class [MWW17b]. For DAGs, these edge relations are:

- a "left-to-right" edge ($X \rightarrow Y$);
- a "right-to-left" edge ($X \leftarrow Y$); and
- neither edge ($X \quad\quad Y$).[2]

When we allow for cycles, there is a fourth relation consisting of both directed edges ($X \leftrightarrows Y$). Here we consider only the three edge relations for DAGs. To fully specify a causal graph over N variables, we need to instantiate exactly one of these edge relations for each of the $\binom{N}{2}$ pairs of variables in the graph. Once a particular edge relation is instantiated for a pair of variables (e.g., $X \rightarrow Y$), there are two other possible edge relations—two degrees of freedom—that the pair can take (e.g., $X \leftarrow Y$ and $X \quad\quad Y$). The trivial equivalence class that contains every possible causal graph (satisfying zero constraints) thus has $2\binom{N}{2}$ degrees of freedom. Note that this number is much smaller than the number of possible causal graphs over the same number of variables.

Each causal graph in an equivalence class instantiates these edge relations differently for at least one of the pairs of variables. For each pair of variables in a system, we can determine the number of instantiations that remain underdetermined by looking at the set of all edge relations

[2] The blank space between the two variables is intentional; it is meant to call attention to the fact that the corresponding nodes in the graph lack any type of edge between them.

that appear in a particular equivalence class. In the example of an equivalence class discussed above, the graphs all agree that there is no edge for the pair $\{X,Z\}$. This edge relation is thus fixed: regardless of which graph is correct, we know what the edge relation for this pair is $X \quad\quad Z$. The graphs in this equivalence class unanimously agree regarding the *existence* of edges for the pairs $\{X,Y\}$ and $\{Y,Z\}$; however, they do not unanimously agree regarding the edges' *orientations*. This equivalence class thus has two degrees of freedom. This metric can be expressed as a percentage to convey the amount of underdetermination relative to the number of variables in the system. Returning to the example equivalence class above, there are $2/(2\binom{3}{2}) \approx 33\%$ of the degrees of freedom remaining. Once enough constraints have been supplied to prune an equivalence class to only one graph, zero degrees of freedom remain. This pruning of the equivalence class thus provides an analytic expression for Popper's conception of science based on falsifiability [Pop59].

2.1.5 Piecemeal causality

Piecemeal causality [May14, May19, May11] is the stitching together of indvidual experiments to find a larger network suitable for fully contextual causal discovery [Ebe13, Ebe10, Ebe09, HEH13]. The piecing together of observational data introduces new sets of challanges [May14, May11], nevertheless discoveries can be made in such context [May14]. The abundance of data plays a vital role in the ability to make meaningful discoveries from piecemeal causality [May11, May19]. The increased data leads to an increase in the amount of unknown variables, to discover the true value of the variables it is suggested to examine every variable [May11], or approximate the values using statistical assumptions [May19].

One technique to PCD is to evaluate the problem under different models and select the most appropriate one [VJB06b, VJM00]. Experimental networks can be considered in examining the various alternative models [VJB06b]. Starting with the inclusion of all possible models, the approach is to focus on eliminating models via experimentation. The work advances that some models may be shown to be superior over others. Model selection is again considered by [VJM00]. This work focuses on a perturbation of competing models via experimentation. Entropy is then used to further refine the result set.

More recent approaches to solve PCD problems are to format the input into a set of constraints and then feed the input into a module that utilizes answer set programming (ASP) [GKK11, Bar03, HEJ14]. In [Bar03], the declarative problem solving approach is shown to be an effective approach to some causal problems. These problems can be synthesized into a reduced declarative programming problem which is then solved by answer set semantics. Related approaches utilize boolean satisfiability solvers that have been used to help aid in causal discovery [HHE13, HEJ14]. One such method advanced by [HEJ14], provides a technique to use boolean satisfiability solvers while accounting for statistical variations of the variables. This approach uses a new logical encoding methodology to obtain high accuracy in causal discovery.

2.2 Topic interpetability and labeling

2.2.1 Topic modeling

Topic modeling began as a subfield of information retrieval with the goal of obtaining short descriptions from a much larger body of text that preserves the essential statistical relations. From the short descriptions the full text can be summarized, compared to other text, classified, and be used in novelty detection. Initially, techniques such as tf-idf vectors were used to summarize a given body of text. [SM84]. To allow for more reduction, latent semantic indexing (LSI) [DDL90] takes the tf-idf vectors and uses singular value decomposition to capture a small set of variance from the input feature set. This technique while achieving the desired reduction, also results in some semantic cohesiveness between the derived features. To allow for more probabilistic reductions of the original textual data, probabilistic LSI (pLSI) [Hof99] was developed which introduced a generative model that models each word as a sample from a mixture model. This mixture model is defined over the vocabulary and can be thought to represent a "topic." Probabilistic LSI was improvement over LSI, however the model lacked a probabilistic model over the set of documents. To allow for this document model, latent Dirichlet allocation (LDA) [BNJ03] was introduced which provided a distribution over the vocabulary (topics) and distribution over the documents. Recently, the field of topic modeling has shifted towards leveraging deep neural networks [Dua21, Che21a, Rez20, Nin20] for topic discovery. These works have shown to be better for predicting held out data and in pointwise mutual information (PMI) based scoring than Bayesian (LDA-based) topic models.

2.2.1.1 Dirichlet Distribution

The Dirichlet distribution is a distribution over probability mass functions with a specific number of atoms and is commonly used in Bayesian models. A property of the Dirichlet that is often used in inference of Bayesian models is conjugacy to the multinomial distribution. This allows for the posterior of a random variable with a multinomial likelihood and a Dirichlet prior to also be a Dirichlet distribution.

The parameters are given as a vector denoted by α. The probability density function for a given probability mass function (PMF) θ and parameter vector α of length J is defined as:

$$f(\theta, \alpha) = \frac{\Gamma(\sum_i^J \alpha_i)}{\prod_i^J \Gamma(\alpha_i)} \prod_i^J \theta_i^{\alpha_i - 1} \tag{2.10}$$

A sample from the Dirichlet distribution produces a PMF that is parameterized by α. The choice of a particular set of α values influences the outcome of the generated PMF. If all α values are the same (symmetric parameter), as α approaches 0, the probability will be concentrated on a smaller set of atoms. As α approaches infinity, the PMF will become the uniform distribution. If all α_i are natural numbers then each individual α_i can be thought of as the "virtual" count for the i_{th} value [Min03].

2.2.1.2 Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA) [BNJ03] is an unsupervised technique for discovering topics in a corpus. LDA represents the basis for many existing Bayesian probabilistic topic models. The topics the model considers are composed of a discrete distribution over the vocabulary. Each document is also assigned a distribution over the set of topics.

LDA is a hierarchical Bayes model which utilizes Dirichlet priors to estimate the intractable latent variables of the model. The corpus (C) is assumed to be generated according to a generative model that samples the mixture of words (words per topic) from a Dirichlet distribution parame-

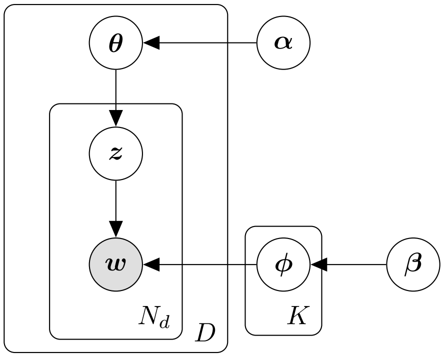


Figure 2.2: Plate notation for LDA.

terized by β and each document (topics per document) is sampled from a Dirichlet distribution parameterized by α. To build a document, after the topic to document mixture is sampled (θ), for each word we sample a topic assignment (z), and for that sampled topic we sample a word from the corresponding topic's mixture over words (ϕ). This is given as:

```
for $k \leftarrow 1$ to $K$ do
    Choose $\phi_k \sim$ Dir$(\beta)$
end for
for $c \leftarrow 1$ to $|C|$ do
    Choose $\theta_c \sim$ Dir$(\alpha)$
    Choose $N_c \sim$ Poisson$(d^*)$
    for $n \leftarrow 1$ to $N_c$ do
        Choose $z_{n,c} \sim$ Multinomial$(\theta)$
        $w_{n,c} \sim$ Multinomial$(\phi_{z_{n,c}})$
    end for
end for
```

With d^* defined as the average length for each document. From the generative algorithm the resultant Bayes model is visualized in plate notation by Figure 2.2.

Bayes' law is used to infer the latent θ distribution, ϕ distribution, and z

$$P(\theta,\phi,z|w,\alpha,\beta) = \frac{p(\theta,\phi,z,w|\alpha,\beta)}{p(w|\alpha,\beta)} \tag{2.11}$$

Unfortunately, the exact computation of this equation is intractable. Hence, it must be approximated with techniques such as expectation-maximization [BNJ01], Gibbs sampling or collapsed Gibbs sampling [GS04]. An example Gibbs sampling update equation is:

$$P(z_i{=}j|z_{\text{-}i},w) \propto \frac{n^{w_i}_{\text{-}i,j} + \beta}{n^{(\cdot)}_{\text{-}i,j} + V\beta} \cdot \frac{n^{d_i}_{\text{-}i,j} + \alpha}{n^{(d_i)}_{\text{-}i} + K\alpha} \tag{2.12}$$

With n being count matrices for word counts in a topic or topic counts in a document, V the size of the vocabulary of the corpus, K the number of topics, w the vector of words in document i and d the vector of documents of the corpus.

2.2.2 Nonparametric topic modeling

Nonparametric topic modeling is based off the hierarchical Dirichlet process [Ble03]. These initial techniques interpret the Dirichlet process as a Chinese restaurant franchise, which is an alternate view of the hierarchical Dirichlet process. Inference can be made in a similar manner to parametric topic modeling—by using Markov chain Monte Carlo techniques. Later techniques have shown inference between non-parametric and parametric topic modeling to be almost the same [WWA21].

A subfield of nonparametric topic modeling is that of hierarchical topic modeling. These techniques seek to find semantically hierarchal topics in a corpus [Ble03, Man21, Che21a]. The models can be based on the Dirichlet process or a similar method such as using a directed acyclic graph [LM06, MLM07]. The generalizations produced by these models have been shown to discover meaningful relations among topics.

2.2.2.1 Dirichlet Process

The Dirichlet process [Fer73] is a probability distribution that describes a method of generating a series of separate probability distributions. Each subsequent probability distribution is generated with a decreasing probability. The process is parameterized by some underlying distribution and a variable or variables that determine when to generate a new distribution.

Though each generated probability distribution is created with a decreasing probability, the process has no stopping criteria. One way to think of the likelihood of generation is to take the previous probability of generating a new distribution and multiplying it by some probability value. This continuous multiplication of probabilities asymptotically tends to zero however will never actually be zero. For this reason, and because the underlying distribution is most often a Dirichlet distribution, the Dirichlet process is often thought of as the infinite length Dirichlet distribution.

To understand the generative process better, several analogies have been created that describe the process. In the Chinese restaurant process, a customer sits at an existing table (assigned to an existing generated distribution) with probability equal to the number of customers already seated at the respective table, or is seated at a completely new table (newly generated distribution) with probability equal to some scaling parameter γ. Another such analogy is that of the stick-breaking process. Each time a distribution is requested, a stick that was initially at length 1 is broken with a percentage length drawn from the beta distribution parametrized by $(1, \gamma)$. Each stick break length represents the probability that the corresponding probability distribution is returned on subsequent requests. The third most common view of the Dirichlet process is the the Pólya urn scheme—where a ball is drawn from a bag, then returned to the bag along with a duplicate of the ball. The bag represents a distribution that resembles the Dirichlet process distribution.

We define a sample from the Dirichlet process as:

$$G_i \sim DP(\gamma \cdot G_0) \tag{2.13}$$

With G_0 being the underlying distribution. It follows then that the posterior takes the form of:

$$P(G|\overrightarrow{G}) \propto DP(\gamma \cdot G_0 + \sum \overrightarrow{G}) \tag{2.14}$$

2.2.2.2 Hierarchical Dirichlet Processes

The concept of nonparametric topic modeling was shown to be possible using a hierarchical Dirichlet process [TJB06]. By doing so, topics were able to be discovered without supplying the number of topics a priori. The solution utilizes the concept of the Chinese restaurant franchise, which is a hierarchical interpretation of the Chinese restaurant process. Inference was done using Markov chain Monte Carlo algorithms. Another interpretation of the Chinese restaurant franchise is the

Chinese district process—whereby a customer is searching for a table inside a number of restaurants in a district [PC09].

A similiar process to the Chinese restaurant franchise, the Pitman-Yor process can also form the basis for hierarchical n-gram models. For example, a hierarchical model based off the Pitman-Yor process is advanced by [Teh06]. In this model, inference is based on an approximation and yields results similar to established smoothing methods.

Another similar model to the Chinese restaurant franchise is the Indian buffet process [GG11]. The underlying distribution of the Indian buffet process has been shown to be a beta process [TJ07]. This interpretation can be useful in document classification and other machine learning tasks.

Outside of traditionally used Markov chain Monte Carlo methods for inference of hierarchical Dirichlet processes [PR08], inference can be done using collapsed variational inference [TKW07]. This technique has advantages over Gibbs sampling and is general to the hierarchal Dirichlet process. Furthermore, other techniques such as slice sampling have also been used in place of Gibbs sampling [TGG07].

Other adaptations of the Chinese restaurant process have been proposed [AX08] which are useful in evolutionary clustering for tree construction [FGM07], connected data across data prevalence [WWH10], and for Pachinko allocation [Li07]. Another area utilizing the hierarchical Dirichlet process is genomic applications where discrete nonparametric priors can be used in place of the previously used priors [LPW08]. Additionally, and adaption of hierarchical Dirichlet processes is demonstrated in labeling topic models which suffer from the problem of interpretability [RMD11].

Since hierarchal Dirichlet processes are a well established and not so emerging technique, much of the new research work does not involve the theoretical interpretation of the process to improve upon the sampling. Recent work focuses more on the application of the hierarchal Dirichlet processes [MAO21, MS21, SSS20, LLF20, CLP21] and less on estimation and sampling techniques [BFP21].

2.2.3 RNN-based topic modeling

The combination of RNN's and topic modeling has been shown to capture semantic dependencies at both the word and document level to improve word prediction [DWG17]. This method, named

TopicRNN, achieves improvement by adding an RNN into the topic generative model. Another technique that involves utilizing latent feature word representations is advanced by [Ngu15]. This generative model mixes the traditional LDA based approach with a maximum likelihood estimation to learn features—and results in improvements in topic coherence and the classification of documents.

2.2.3.1 Recurrent Neural Networks

Recurrent neural networks (RNNs) are a class of neural networks well suited for sequential input. They are straightforward extensions of the standard feedforward multilayer perceptron networks [Ros61], which add a cycle to the hidden layer. At each time step, the RNN updates its hidden state per the new input. The RNN's hidden state is thus maintained and passed through time, which can distill information from an infinite-length context window. Such mechanisms can endow RNNs with long-term learning, which allows them to model long distance dependencies effectively.

Given a sequence of input vectors $(x_1, ..., x_T)$, the RNN predicts the output sequence $(\hat{y}_1, ..., \hat{y}_T)$ using the following equations:

$$h_t = tanh(W^{hx}x_t + W^{hh}h_{t-1} + b^h) \tag{2.15}$$

$$\hat{y}_t = softmax(W^{yh}h_t + b^o) \tag{2.16}$$

where h_t is the high-dimensional hidden state at the time-step t, W^{hx}, W^{hh}, and W^{yh} are the weight matrices connecting the input layer to the hidden layer, the hidden layer to the hidden layer, and the hidden layer to output layer, respectively, and the vectors b^h and b^o are the biases.

One problem with traditional RNNs is the vanishing gradient. Because each time step of the RNN is a product of the input and each previous time-step hidden layer, on back propagation the distant axons are multiplied by a smaller and smaller value. This leads to an increasingly low or even nonexistent influence onto the current time-step's edge weights. A popular solution to mitigating the vanishing gradient problem is the Long Short-Term Memory (LSTM) neuron. These neurons replace the simple hidden layer neurons of the traditional RNN with a more complex gating neuron which regulates the it's inner state to forget unimportant information and remember important information.

2.2.3.2 Deep Learning Sequence Generation

Deep learning sequence generation research has successfully injected side information into RNN models on a word-level basis to achieve increased performance [Hoa16]. Hoang et al. analyzed different methods of subjecting information into the input, hidden, and the output layer of the RNN language model (RNNLM). The research shows and effective and consistent performance boost to the non-sub structured RNNLM.

One drawback to the RNNLM is that long-range information cannot be forwarded between phrases. To improve on the basic RNN structure, recent research has looked into combining the LSTM with the RNN to generate long sequences [Gra13]. This work demonstrates the possibility of generating complex long-range sequences. Additionally, the method successfully extended the approach to a handwriting synthesis, where they could condition the text sequence to a highly realistic handwriting style.

Kim et al. demonstrated an approach to predicting next characters in text streams by using a large RNN-powered by a Hessian-Free optimizer [SMH11]. Similarly, character-level generation was shown to be attractive in achieving the asymptotic limit of text compression [War00]. While word-level generation is limited to the vocabulary set of the training set, character-level generation can generate an “understanding” using the intelligence of the model.

Character-level generation is investigated using other models as well, such as the model proposed by Santos et al., CharWNN, that attains state-of-the-art results for a language-independent model [SZ14]. Research by Zhang et al. demonstrates a deep character-based CNN network performs measurably well on text classification problems. Additionally, this work shows that character-level generation can be done without structure if several conditions are satisfied [ZZL15]. These conditions are: the size of the dataset, if the texts are curated, and the right selection of the character sets.

2.2.3.3 Topic and Document Modeling RNNs

Utilizing deep neural networks has alleviated several document classification challenges in research today. Tang et al. developed a neural network model for sentiment classification[TQL15]. The results over the IDMB and the Yelp Dataset Challenge have achieved performance improvements over state-of-art algorithms and the current sentiment classification model using a neural network. In the first step, the model uses a CNN or an LSTM with a word model to create new sentence models. In the second step, a gated RNN translates relationships between sentence semantic and the document models. In their result, they have found that the LSTM achieved a higher performance compared to a multi-filtered CNN.

Wan et al. demonstrates the use for topic modeling in neural networks on image classification [Wan]. The image data contains scenic images such as an office, bedroom, living room, etc., a total of 15 scenic categories. The experiments were designed in three different settings: hierarchical topic modeling, a neural network, and a hybrid model that combined both the hierarchical model and the neural network. Throughout the experiment, the hybrid model outperforms the neural network by a wide margin and achieves comparable results against the supervised LDA model.

2.2.4 Supervised topic modeling

Supervised Latent Dirichlet Allocation (sLDA) is a supervised approach to labeling topics [BM07]. The approach includes a response variable used in the LDA model to obtain latent topics that potentially provide an optimal prediction for the response variable of a new unlabeled document. The model requires the manual input of individual topic labels and is constrained to permitting one label per topic.

Similar to sLDA is Discriminative LDA (DiscLDA) which attempts to solve the same problem as sLDA, but differs in the approach [LSJ08]. The differing approach was centered around introducing a class-dependent linear transformation on the topic mixture proportions. This transformation matrix is learned through a conditional likelihood criterion. This method has the benefit of both reducing the dimension of documents in the corpus and labeling the lower dimension documents.

Both sLDA and DiscLDA only allow for a supervised input set that label a single topic. An approach that allows for multiple labels in a topic is given by Labeled LDA (L-LDA) [RHN09]. This model differs in the generation of the multinomial distribution, theta, over the topics in the model. The scaling parameter is then modified by a label projection matrix to restrict the distribution to those topics considered most relevant to the document.

2.2.5 Semi-supervised Topic Modeling

A known weakness of traditional topic modeling (LDA) [BNJ03] is a lack of interpretability [CBG09]. Given a set of topics, a human annotator can often have difficulty identifying a label for the topics. One solution to increase interpretability comes in the form of semi-supervised topic modeling.

Some forms of semi-supervised topic modeling combine unsupervised models with supervised models to classify documents [Aga21, LML21]. Although a model could be adapted to utilize a knowledge source [PD21], the supervised portion would still require an input set that may be expensive or challenging to obtain. Other forms of semi-supervised topic modeling take the semi-supervised input directly from the input itself [SRR21]. While the approach is effective, such as in the domain of sentiment classification, this may not apply all domains since they do not incorporate outside knowledge and may lead to problems when the input does not contain enough information to form meaningful topics [WTW17].

2.2.6 Weakly-supervised Topic Models

Weakly-supervised topic models are a subclass of topic models that are mostly extensions of the Dirichlet-based Bayesian topic model, latent Dirichlet allocation (LDA) [BNJ03]. The desideratum of Weakly-supervised models is to utilize a large collection of documents that have already been labeled to improve topic labeling and interpretability. These documents are formed into topics and serve to bias some subset of the existing LDA topics. The set of labeled topics (extracted from the labeled documents) is often referred to as a *knowledge source*. We formalize the *knowledge source*

(KS) along with the associated topic distribution ($\hat{\phi}_j$) as:

$$\hat{\phi}_j \sim f_\phi(\hat{X}, KS, \hat{A}_j, \beta) \tag{2.17}$$

$$KS = (\hat{A}_1, \hat{A}_2, \ldots, \hat{A}_{\hat{B}}) \tag{2.18}$$

$$\hat{X}_j = f_{\hat{X}}(\hat{A}_j) \tag{2.19}$$

$$\hat{A}_j = (\hat{L}_j, \hat{w}_{1,j}, \hat{w}_{2,j}, \ldots, \hat{w}_{\hat{G}_j,j}) \tag{2.20}$$

With $\hat{G}_j$ being the word count of article-topic $\hat{A}_j$, $\hat{L}_j$ the article label, $\hat{w}_{i,j}$ as the ith word in $\hat{A}_j$, β is a Dirichlet distribution hyperparameter and f being a function determined by the model.

This labeled topic input is assumed to be part of the generative model. Before generating the corpus, we determine the total number of topics (K) and vocabulary size (V). For each *topic*, we sample from a Dirichlet distribution that may or may not be influenced by an individual knowledge source topic. If a knowledge source topic influences the topic, the *topic label* becomes the article's title from which the knowledge source topic was created ($\hat{L}$). Each *document* in the corpus is generated by first sampling a *topic* from a discrete distribution of size K. After the topic is sampled, a *word* is chosen by sampling from the topic's discrete distribution (ϕ) of size V.

During the corpus generation, some topics are formed using the technique from LDA, while a set of others are drawn from a function of the labeled input data. This function can place a Dirichlet prior over the vocabulary [SSC11, PS20]; however, this tends to lose any semblance from the labeled input data. Since the labeled input data tends to be highly interpretable, a more interpretable approach involves assuming the labeled input data are topics themselves [Han13, ND18]. In these approaches, the labeled input often comes in the form of documents describing a topic, which becomes the knowledge source. The labeled documents are formed into histograms and directly turned into distributions representing the histograms. While these approaches increase interpretability, they can be too rigid in representing a labeled topic, leading to a decrease in the number of interpretable labeled topics [WTW17].

An example plate diagram for a weakly-supervised topic model is highlighted in Figure 4.15, with the variables explained in Table 4.9 and Table 4.10. Since the labeled topics are part of the generative model, the inference must consider these new variables for any weakly-supervised topic model. Weakly-supervised methods generally assume a modification to only that of the ϕ-distributions. Therefore, when building a Gibbs sampler each model considers some existing posterior density calculation (for example the posterior density used in LDA) alongside a posterior density that utilizes a predetermined knowledge source. A general Gibbs sampler [GS04] can be built using the sampling condition given as:

$$P(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i},\overrightarrow{w},x,y) \propto f_L(x,i,j,n_z,n_d,\beta,\alpha,K) \tag{2.21}$$

which is the traditionally used posterior density (f_L) that can take on different forms [GS04, Wal08] for $j \leq K$. For $j > K$ we use the knowledge source density of:

$$\hat{T}_j = (\hat{L}_j, \hat{w}_{1,j}, \hat{w}_{2,j}, \ldots, \hat{w}_{\hat{V}_j,j}) \tag{2.22}$$

$$\hat{X}_j = t_{\hat{X}}(\hat{T}_j) \tag{2.23}$$

$$P(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i},\overrightarrow{w},x,y) \propto f_S(x,i,j,n_z,n_d,\beta,\alpha,K,\hat{X},\hat{T}) \tag{2.24}$$

where: $\overrightarrow{z}$ is a vector of topic assignments for document x, i is the index of the current token in document x, $\overrightarrow{w}$ the vector of words for document x, n_z is the count matrix for each word and each topic, n_z is the count matrix for each topic and each document, β is the symmetric hyperparameter for the word to topic mixtures, α is the symmetric hyperparameter for the topic to document mixture, K is the number of all non-labeled topics, y becomes the index of w_i, $\hat{L}_j$ is the label associated with knowledge source topic j, $\hat{w}_{1,j}$ is the count of word w_1 in knowledge source topic j, V the size of the vocabulary, with $t_{\hat{X}}$ and f_S as a transformation and density function specific to the model.

2.2.7 Topic modeling rankings

The original PageRank algorithm was used to determine the importance of a given web page [PBM99]. This ranking was shown to be an excellent way to rank the importance of a word in a sentence [MT04] and determine trustworthy websites [Gyo04]; however, they are only applicable to their respective domains. The integration of topics into PageRank can help weight the edges used in PageRank. This weighting is accomplished using topic vectors determined by topic lists [Hav02]. PageRank has also been used to add weighting to some classification and topic tasks [AHB20, LC21, JDR20, BRD20]. As in TextRank, vanilla PageRank works to find an importance ranking among connected components. These and other approaches help form a motivation for our goal of improving weakly-supervised models, but they do not offer insights on how to integrate PageRank with weakly supervised topic models.

2.2.7.1 PageRank

The PageRank algorithm [PBM99] was developed to objectively determine the best web sites that human users pay attention to. The method consists of web pages filled with links. A link from a website A to another website B was modeled as two nodes for each website (A and B) and a directed edge connects node A to node B. The world wide web could then be modeled as a directed graph.

To determine a ranking for a node, the intuition was to give high importance to the node if it had many other highly important nodes pointing to it. This implies an iterative algorithm where at each step, the current ranking is calculated given the current state of the model, and continued until convergence.

Additional variables are added to the single step calculation due to the potential for some nodes to have no outbound edges, or no inbound edges. This leads to the calculation of a single step ranking (R) for a given node m as:

$$R(m) = \frac{1-d}{|N|} + d \times \sum_{n \in I(m)} \frac{R(n)}{|O(n)|} \tag{2.25}$$

where $I(x)$ and $O(x)$ are functions that returns a set of all the inbound and outbound nodes connected to node x respectively, and N is the set of all nodes. The variable d represents the damping factor, which can be interpreted as a probability to mitigate against those nodes which have no inbound or outbound edges.

An interpretation of PageRank can be that of a random surfer. The random surfer starts at any one web page randomly in the world wide web and either clicks a random link with a probability of d or goes to any other page randomly in the world wide web. The result of the ranking is the probability that the random surfer will visit a particular page.

2.2.8 Topic Labeling

Topic labeling can be done in the post-processing stage [MSZ07, LGN11] by comparing the topic distributions with some predetermined knowledge source. The drawback of these approaches is that the topics tend to cluster non semantically related words [WTW17]. Prior to post-processing labeling, labels were often generated by hand [MSZ07, Mei06, MZ05, MZ06]. Though manual labeling may generate more understandable and accurate semantics of a topic, it costs a lot of human effort and it is prone to subjectivity [WM06]. An example of manual labeling is given by the Topics over Time (TOT) model [WM06]. This method implements continuous timestamps with each topic [WM06], and is shown to produce accurate topics along with more accurate time stamp predictions.

Mei et al. proposed probabilistic approaches to automatically interpreting multinomial topic models objectively. The intuition behind these algorithm was to minimize the semantic distance between a topic and a label. To this end, they extracted candidate labels from noun phrases chunked by an NLP Chunker and the most significant 2-grams. Next, they ranked labels to minimize Kullback-Leibler divergence and maximize mutual information between a topic and a label. The approach achieved the automatic interpretation of topics, but available candidate labels were limited to phrases inside documents.

Lau et al. developed an automatic topic label generation method which obtains candidate labels from Wikipedia articles containing the top-ranking document terms, titles, and sub-phrases. To rank the candidates topic labels, they used different lexical measurements, such as point-wise mutual information, Student's t-test, Dice's coefficient and the log likelihood ratio [Pec10]. Supervised methods like support vector regression were also applied in the ranking process. Results showed that supervised algorithm to outperform unsupervised baseline in four corpora.

In previous approaches, topics were treated individually and the relationships among topics was not considered. Mao et al. created a hierarchical descriptor for topics, and the results proved that inner-topic relations can increase the accuracy of topic labels [MMZ12]. Topic relationships were also considered by [HHK13] in their graph-based approach to topic labeling. In a similar manner, Mehdad et al., build an entailment graph over phrases—and from the entailment graph, relevant phrase were aggregated by generalization and merging [MCN13].

Conceptual labeling is an approach to generate a minimum sized set of labels that best describe a bag of words which includes topics generated from topic modeling [SXW15]. Concepts used in the topic labeling are taken from a semantic network and deemed appropriate using the introduced metric: Minimum Description Length. This approach is applied after topic modeling and represents an effective way of labeling topics.

Supervised approaches allow predetermined labels to be assigned to clustered topics [BM07, LSJ08, RHN09]. These approaches can assign an entire document a label [BM07, LSJ08] or assign multiple labels to a document [RHN09]. Supervised techniques are often dependent on an extensive collection of labeled data that may be expensive or time-consuming to obtain.

A balance between after-inference topic labeling and supervised topic labeling comes from weakly-supervised topic modeling, previously discussed in Section 2.2.5. These methods use some form of labeled input, much like supervised topic modeling; however, the input is much easier and cheaper to obtain.

More recent approaches use deep neural networks to label topics or perform similar tasks [Gho21, YHC20]. One example is mapping the vectorized tokens against the tokens of the corpus [Gho21]. After the corpus tokens are mapped to vectors, classification is run using a deep learning model. This approach yields good results when the input is labeled and enough training data exists to build a supervised model. When there is not enough labeled data, the supervised model may yield poor results.

2.2.9 Contextual Integration

An example of other forms of contextual integration is given by [SSC11]. The approach is constructed by taking into account concepts supplied by prior sources and requires a manual input set of relevant terms. The authors then integrate this context into a topic model and the concepts (context) are applied to the assignment of tokens to a topic. Alongside this concept topic modeling a hierarchical method can also be used to incorporate concepts into a hierarchical structure. This work shows the utility of bringing in prior knowledge into topic modeling.

Other research related to the incorporation of concepts into topic modeling are demonstrated in [Han13]. This work brings prior knowledge in the form of Wikipedia information into the topic model. Additionally, the model only requires an existing Wikipedia article, as opposed to manually curated data often used in supervised topic modeling. The assumption of the model is that in the generative process the topics are selected from the Wikipedia word distributions—and the results show that Wikipedia articles can be used as effective topics in topic modeling to improve interpretability [Han13].

Wikipedia again was shown as an enhancement for topic modeling, albeit for a tangential approach, entity disambiguation [HC13]. The approach involved topic modeling as a way of annotating entities in text. This involved the use of a large dataset of topics requiring the development of efficient processing methods. Experiments against a public dataset resulted in a state of the art performance.

2.2.10 Interpretable topic modeling

The interpretability problem in topic models was established by asking humans to find relationships among words comprising topics [CBG09]. This work provides a technique to score the interpretability level and presents findings that show topic models are not highly interpretable [CBG09]. With this deficiency established methods have been developed to increase the interpretability of topics. Existing methods can use visualization, careful selection of displayed words, or interacting periodically with annotators to increase semantics [DGW20, PMM21]. Recent methods seem to have shifted toward neural based topic models and represent the state-of-the art approach for obtaining high pointwise mutual information (PMI) based scores [New10, Dua21, Che21a, Rez20, Nin20, Bia21, Tom20].

2.3 Contributions of this dissertation

2.3.1 Causal extraction using alignments

Given the work achieved in ResearchMaps to date, we have a small, limited amount of labeled data to form our causal graph. However, we can expand the number of causal discoveries from this small, labeled data set by matching the textual input to a vast set of candidate fragments. We introduce a novel technique for matching text fragments to accomplish this goal. The algorithm is based on sequence alignments and expands on previous work by allowing two sequences to be broken into a variable number of breakpoints and then aligned. This work (1) serves to improve the capabilities of existing sequence alignment algorithms and (2) represents a novel technique to piecemeal casual extraction. With our developed algorithm, named OpBerg, we bestow upon the research community another tool in the search of causality. We also form the foundation for our textual extraction technique by which we can increase the number of causal connections in our understanding of the causal world.

2.3.2 The piecemeal cumulative evidence index (PCEI)

In describing piecemeal causality, it is vital to quantify the strength of evidence. Furthermore, any score must agree with the concepts of convergency and consistency [MWD18]. Convergency represents the same underlying structural result between experimental participants under different conditions—while consistency is the same result when performing the same experiment. To this effect, we developed a Bayesian model [Kra17] that amalgamates the heuristics of the biological world and the field of statistics. The result of our novel method referred to as the piecemeal cumulative evidence index, allows us to describe the strength of evidence of a piecemeal causal assertion. This quantification simplifies the concepts of convergency and consistency into a numerical form and contributes to the field of meta-research [IFD15]. The metric can help scientists determine which experiments to focus on and can help deduce which experiment to perform next.

2.3.3 Causal discovery from textual piecemeal casual statements

Unlike data-driven causal discovery we develop a technique to discover causal from textual statements assumed to represent an individual causal statement. The individual causal statements are then stitched together to form piecemeal causal networks from which causality can be discovered. The techniques are congruent with data-driven causal discovery and serves to enhance the overall understanding of causality. This meta-analytic approach helps researchers further understand a more robust causal world independent of inferential data. The only necessary component is a textual assertion describing the causal phenomena. Given the simplicity of our technique, we can discover piecemeal causality from a robust amount of data sources and corpora. This approach expands the scope of causal discovery to a vast amount of data. From the textual assertion, we connect the existing elements connected in other assertions. This causal structure, representing a graph, can be used in existing causal discovery algorithms [HEJ14] to uncover a true causal graph among competing evidence. Next, given our set of causal networks consistent with the evidence at a given time, t, we can then proceed to make assertions about the causal structure until we have a complete understanding of the causal network.

The graphical nature of our piecemeal representations is beneficial to scientists in how the causal assertions are synthesized into a common form well situated for knowledge discovery. The format is made in such a way that constrains the empirical findings to help the scientist differentiate between causal facts and background assumptions. The delineated background assumptions can then be synthesized into formal constraints to add further knowledge to the causal graph [Ebe17]. The background assumptions can be interchanged with different yet competing background assumptions that, together with the causal fact constraints, present the scientist with a synthetic environment in which to test hypotheses. Another benefit allows the scientist to examine whether a test hypothesis is logically consistent with previously established results. The existing state of the causal representation aids future work by allowing the new hypothesis to be tested for consistency. Depending on the determination of the hypothesis, the scientist can be more confident in performing the experiment that confirms the hypothesis, or in contrast, seek out a more congruent hypothesis if the consistency test is conflictive. This process represents a powerful new tool to aid the scientist in planning experiments, a technique that is both time and cost-effective. The causal network represents the synthesis of previously learned information, allowing the scientist to make a more informed decision.

2.3.4 Experiment selection heuristics

A key component of scientific reasoning is uncertainty in an existing scientific setting. We provide a set of heuristics that can quantify the amount of uncertainty, allowing the scientist to convey and ultimately act on this uncertainty amount. This dissertation introduces and submits a novel application, ResearchMaps, to visualize and represent the causal evidence numerically using the piecemeal cumulative evidence index. The score, representing the inherent consistency and convergence, allows the scientist to select an experiment based on the established certainty and uncertainty of an experimental setting. The ResearchMaps visualizations come in the form of graphs that are easily deduced into causal graphs, which present the researcher with causal biological pathways when chained together. Given the evidence, we then provide a technique to aggregate the existing causal world into a set of possible explanations. We introduce a degree-of-freedom metric and an expectation metric to guide the process of experiment selection and reduce the process into the elimination of inconsistent causal networks. We then provide heuristic algorithms that strengthen the natural intuition of the biological researcher. The technique contributes to the scientific process by formalizing and quantifying the process behind selecting an experiment.

2.3.5 Contextual piecemeal causality

So far, the contributions in the domain of piecemeal causality have been made upon two direct participants in most experiments—that of an agent and target. The time and cost difficulties in conducting experiments often limit the scope of investigation to one biological concept acting on another. We add to this environment the surrounding context of the experiment. Given that we have constructed a partial causal network, we develop a scoring approach that quantifies the effect that other variables may have affected the observed outcome. Much like the PCEI, we can synergize with consistency and convergence in numeric form. This additional consideration adds to the causal network even more information. We apply this technique to that of an experimental setting to show that contextual information converges more quickly than that of analyzing the causal network in isolation. This theoretical finding lays the groundwork for real-world applications which we show have the potential to make key scientific findings—such as helping scientists discover the next experiment to find a previously unestablished connection, to find the biological elements most likely to be causing an effect, and which previously established connection is most likely to be due to confounding variables. The addition of contextual assumptions yields two potentially impactful concepts in the piecemeal causal framework: *piecemeal causal decay* and *potential piecemeal causality*. These concepts are expressed in a Bayesian model and then used to make improved discoveries in the causal network. We add into our piecemeal setting the surrounding causal network information for an even richer quantification of piecemeal causality.

2.3.6 The piecemeal causal pipeline

Having established an approach starting from extracting causal text fragments from raw text and ending with discovering potential assertions about the causal world, we connect each of the steps to form our piecemeal causal pipeline. We show the steps and processes needed to arrive at our end discovery, representing a systematic and highly informed process by which the scientist can arrive at hypotheses. The hypotheses can then be tested and fed back into the pipeline arriving at an even more lucid understanding of the biological elements. We demonstrate the applicability of our method by running the pipeline from an exhaustive dataset. From the output of our pipeline (with the input being the exhaustive dataset), we arrive at various discoveries about the existing biological landscape. These include: establishing new causal pipelines, highlighting the amount of certainty about established relationships, key biological elements which are likely to have a causal relationship, known associations that are most likely due to confounding variables, and the most important entities in a causal cluster. Put together this pipeline represents a potentially highly transformative approach to the scientific method—one in which previously established information is accumulated at a massive scale to make a Bayesian-based decision on what steps to take that are consistent with and strengthened by the massive known and documented causal universe.

2.3.7 Foundations for interpretable topic modeling

Topic models cluster words into classes that represent topics in a corpus. The clusters form a distribution based on the number of times each word is assigned to a topic. Additionally, we can assign each document a distribution over each topic based on the topic assignments. To help with the identification of a topic, a label can be placed over the topic distribution that summarizes the allocation of words. To assign a single topic label implies some coherence among the word's assignments to the applied topic label. However, given the naiveness of the Bayesian model, many words are inconsistent with each other. This lack of interpretability restricts topic models from being utilized in applications such as summarizing patient free-text medical records to assist the clinician in understanding the context of a patient. In this dissertation, we contribute to the fields of topic interpretability and topic labeling by introducing our novel method, Source-LDA. We establish a technique to label topics by integrating outside knowledge sources into the Bayesian topic model. The knowledge source integration also helps shape the distributions to be more like the input knowledge source. Since the input knowledge sources tend to be highly interpretable, the topic clusters become highly interpretable as well. Source-LDA lays the foundation for this novel approach to topic interpretability and labeling.

2.3.8 Improvements to weakly supervised topic models

Source-LDA represents a contribution to a larger field of weakly supervised topic models. This subfield introduces into the generative model outside information in order to help shape the topics to be more interpretable. Additionally, a label can be given to the topic clusters based on which outside item was utilized to form the cluster. However, the models are limited to the amount of input that can be used to help bias the topics. This limitation is due to the inference method, which requires the input to be iterated over in conjunction with each token in the corpus. To mitigate this problem, we introduce an approach to filter out noisy topics a priori. Our approach utilizes PageRank and eliminates input while maintaining the benefits of the weakly supervised topic models. This addition allows weakly supervised topic models to be used with a 100-fold increase in input size. The benefits of our novel ranking method are extended to the topic model inference, which improves the interpretability and perplexity of weakly supervised models.

We also seek to improve upon our weakly-supervised input. We add the deep learning recurrent neural network (RNN) into our weakly supervised approach. Given the ability of RNNs for word and character prediction, we utilize this ability to assist the weakly supervised topic model in predicting words outside the weakly-supervised input. By doing so, we can improve our topic assignments based on the words' structure. The RNN is trained against the input set, and each unknown word is asked which trained input it belongs to. This approach takes advantage of a strength of the RNN to improve a weakness of the weakly supervised model. This strategy drastically improves the topic model assignments, further increasing the interpretability. Remarkably, this relationship also works in reverse as we contribute to the field of RNNs by adding an ensemble method partitioned by topic. Our technique, referred to as Topic-RNN, represents the state-of-the-art addition of topic modeling into the RNN.

2.3.9 A novel view of the Dirichlet process

The Dirichlet process generalizes the Dirichlet distribution over an infinite dimension. The continuous nature of the stochastic process makes the Dirichlet process appealing in the topic modeling setting. Assuming a generative model constructed by the Dirichlet process allows for corpora to theoretically contain an infinite number of topics. Additionally, the need to specify the number of topics a priori is unnecessary, and the variable is removed from the input set. The result is a more flexible and a potentially more correct discovery of topics. To help in understanding and deriving inference approaches to the Dirichlet process, alternative views are established which include the Chinese restaurant process, the Pólya urn scheme, and the stick-breaking process. Each view has a representative and equivalent derivation for inference. In this dissertation, we present a fourth alternative view that is quite suitable for topic modeling. The process assumes a bank teller partitioning a table full of coins by biasing the coins to a distribution for each partition, then flipping each coin and placing the coin into the partition if the coin lands on heads. This interpretation, called the biased coin flip process, results in an inference closely resembling existing topic modeling inference equations. The similarity results in the applicability of the Dirichlet process into topic modeling that produces state-of-the-art perplexity and topic number discovery. The view also serves as a more straightforward bridge to understanding non-parametric topic modeling.

2.3.10 A complete topic model for interpretability and topic labeling

Source-LDA provides the foundation for interpretable topic modeling and labeling. However, a few shortcomings exist that limit the applicability of the model. The main drawback is the relatively limited input size that can be used in the model. The increase in execution time limits the knowledge source to be in the order of 10^3. Our ranking method previously discussed helps in this regard; however, improvements may still be attainable. We present a solution that represents a complete non-parametric topic model that labels topics and is highly interpretable. The approach is an amalgamation between weakly-supervised topic models and non-parametric topic models. We take the inputs of a weakly-supervised input and aggregate out the external variables, which allows us to re-use existing inference equations of non-parametric models. We can also take an approximation of probability for the weakly-supervised input to sample from a higher-order knowledge source input size. Given the foundations we have established in this area, we show the result of a combination to be highly fruitful. The resultant topic model represents a complete non-parametric, highly interpretable topic model, which additionally results in topic labeling. The non-parametric nature of the model allows for an input size that was previously infeasible. A further reduction using our ranking method alongside existing information retrieval techniques allows for a virtually unlimited input size. The model itself takes as input only hyperparameter settings alongside a corpus and results in topics assigned meaningful labels that are highly interpretable. In our evaluations, the model represents the state-of-the-art approach for interpretable topic modeling.

CHAPTER 3

Piecemeal Causality

3.1 Introduction

The number of scientific research articles are expanding at an exponential rate. With such a vast amount of information it is even more difficult for the research scientist to synthesize the information relevant to their interests. One way to reduce the information overload is to extract out the most relevant parts. In biology, the most relevant parts take on the form of causal statements. Given the unmatched ability of computers to index, retrieve, and process information, biologists could benefit enormously from a computational approach to help them to track and reason through causal assertions; such a tool could help biologists to synthesize empirical findings and plan future experiments.

The general need to mitigate this information overload is discussed in [LS13, SM15]. An application stemming from these general overviews is represented in our proposed approach, *research maps*, designed to help biologists integrate and plan experiments. A research map graphically represents hypothetical assertions and empirical findings. The graphical representation is accompanied by a Bayesian calculus of evidence that allows researchers to formally synthesize empirical results. Our approach includes integration principles, including convergence and consistency, commonly used by many biologists to judge the strength of causal assertions. Thus, our goal with research maps is not to build another *ontology* but rather to formalize aspects of biologists' *epistemology* [MWW17a].

Biologists traditionally find research summaries in reviews and opinion articles. Although these articles are useful, they have clear limitations: they are not dynamically updated; it is cumbersome to personalize them; and they usually reflect the state of a field as it existed at least one to two years before the publication date. These limitations are particularly a problem in rapidly changing fields like neuroscience. By comparison, digital lab notebooks are extremely useful for tracking and sharing experiments and findings between collaborators, but they are not designed to track large amounts of causal information in a representation that facilitates evidence synthesis, knowledge discovery, or causal reasoning.

Existing representations such as Knowledge Engineering from Experimental Design (KEfED) provide a way to model experimental procedures and findings in a detailed and machine-readable manner [RRH11]. Here, we present a complementary approach for representing and querying high-level assertions that characterize connections among phenomena. Moreover, formalisms such as probabilistic graphical models (e.g., Bayesian networks) have been shown to be effective at conveying relations among biological phenomena using a graph structure, as they compactly encode the joint probability distribution across variables [Fri04]. However, conditional probabilities are often missing in reports of experiments designed to test causal assertions.

A research map is a representation of an assertion in the biological domain. From the research map we can further synthesize the information down into casual assertions; since a biological assertion contains an implicit causal assertion we can easily extract out the causal implications from a documented biological relationship. The reduction of the research map leads us to knowledge discovery along the steps of the scientific process. From a set of causal assertions, connected together in a piecemeal fashion, we can recommend to the scientist a set of experiments to perform to yield a more informative result. Additionally, we show that we can make assertions about what experiments are likely to be established via experimentation, what results are most likely due to confounding variables, and which entities are most important in a network of causal connections. The scientific method becomes a pipeline with the input being a set of causal statements and the output being knowledge discovery based on a network of piecemeal components. However, the current database of causal statements in the biological domain is somewhat limited. To increase this set, we focus on a technique to extract causal statements from a large corpus of biological research documents.

3.2 Piecemeal causal extraction

Researchers who perform biological experiments convey their discovery in published research articles, which contain descriptions of causal relations. This growing literature provides an enormous amount of information and represents the current state of biological understanding. This documentation of scientific discovery can verify previous experiments, provide insights to researchers [SM15], and motivate future research [Mat17].

Because these corpora of biological text are growing at an exponential rate, algorithms and approaches are thus needed to extract the relevant information, allowing biologists to understand and connect biological processes. Since researchers describe causal connections among biological entities in free-text research papers, it is logical to extract these connections using natural language processing (NLP).

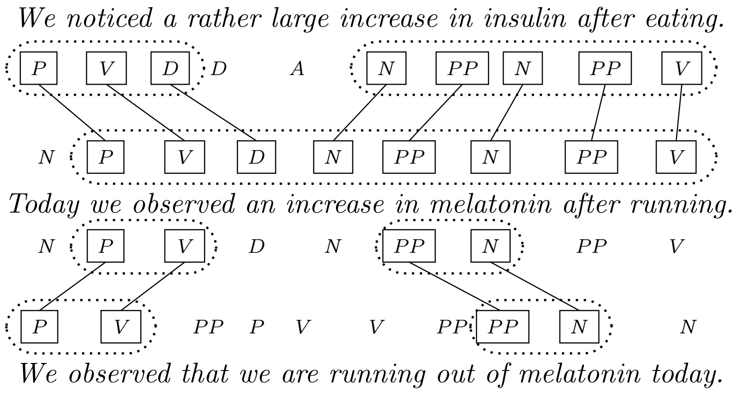
A causal assertion can be thought of as a relation between an agent and a target. Often in biological studies, an agent is either passively observed or actively manipulated, and a change or lack thereof is noted in a target. Although this type of result can be described across many different and sometimes nonadjacent sentences, this paper focuses only on causal assertions appearing in a single sentence. This approach has the advantage of limiting the search range for descriptions of causality and takes advantage of existing methods that can reliably fragment documents into collections of sentences [Man14].

Existing methods for causality extraction use either predefined knowledge bases, word lists, other types of databases [PB21, Che21b, KB91, Sah21, GM02, Wan21, Bui10, Hus21], or are based on statistical techniques—often some form of machine learning [TJ97, LDS21, Gir10, BG21, Do11, Han21, CC04, Fis21]. Predefined knowledge bases are of course limited by the quality of the knowledge base itself. Often, these sources are manually curated and do not always contain all possible words or phrases of interest. Additionally, they require exact matches to be useful. For instance, if a knowledge base contains causal verbs and a potential causal sentence contains the misspelled verb "cuases" (instead of "causes"), the sentence will be dismissed due to the misspelling. These predefined knowledge bases are also not able to capture new words or concepts, and they are not extensible to other tasks such as extracting causality from text in other languages.

One solution to these problems is to use existing machine learning techniques. But these approaches often require large amounts of labeled training data, something that can be expensive and tedious to obtain. These barriers of time and cost are expanded when the task is to discover more fine-grained details pertaining to causality, such as that of finding the specific types of studies and outcomes that lend evidence for a causal assertion. Additionally, the vocabulary for biomedical free text can be quite large, as it contains not only common words but also domain-specific terms. This large vocabulary set requires an even larger training data for the machine-learning model to predict the necessary components for representing causal phenomena.

Thus, to automatically extract causal sentences, an approach is required that does not suffer from limitations in the size of the training data, and that can be performed efficiently. The approach presented in this paper is inspired by the analogy of the aforementioned problem to that of comparing a set of genomic sequences in bioinformatics.

Though it may not be obvious, there is indeed a connection between aligning sequences in genomic data and finding causal sentences in free text. While each sentence may contain a unique set of words, the part-of-speech (POS) sequence of each sentence is likely to be much more common. Breaking each sentence into its grammatical structures can thus help to identify patterns in the way that causal relations are described. Thus, applying an alignment method to the grammatical structures of sentences has the potential to discover similarities that may be missed by approaches that focus only on words. We further illustrate this with the following example of three sentences and their corresponding POS mappings (for brevity we replace the POS label with a single character: P = pronoun, V = verb, D = determiner, A = adjective, N = noun, PP = preposition):


We noticed a rather large increase in insulin after eating.
P V D D A N PP N PP V
N P V D N PP N PP V
Today we observed an increase in melatonin after running.
N P V D N PP N PP V
P V PP P V V PP PP N N
We observed that we are running out of melatonin today.

Here the first two sentences are talking about two different things; yet both are causal sentences. Their POS structures are similar. In comparison, the second and third sentence share a lot of words, more so than the first and second sentences, yet their POS representations have fewer matching elements, with long gaps in between matches. Therefore, knowing that the second sentence is causal, we cannot determine whether the third sentence is causal. It is our hypothesis that given a labeled set of causal sentences $C+$ and non-causal sentences $C-$, a new sentence s is classified as a causal sentence if its POS structure is most similar to a causal sentence and the similarity (S) is above a threshold δ,

$$\max_{c \in C+} S(c, s) > \max_{c \in C-} S(c, s) \wedge \max_{c \in C+} S(c, s) > \delta \tag{3.1}$$

Our desired approach is to find causal relations by comparing the POS mappings of unlabeled sentences to that of labeled sentences. A new causal sentence is discovered by identifying the optimal number of alignments between the grammatical representations of the sentences. We show this alignment approach can thus classify causal sentences accurately and efficiently, and it has the potential to be used for other problems as well.

Existing methods of sequence alignment are insufficient for aligning POS representations of free text: either (1) they require the user to specify the number of local alignments [AG11] or (2) they introduce a gap penalty for each new local alignment [HC03], possibly leading to erroneous alignments [AG11]. Given the nature of free text, it is unreasonable to ask the users to pre-specify the number of local alignments. Here, we generalize existing alignment algorithms by removing the need to specify these parameters, while keeping the same algorithmic complexity in terms of both space and time. This generalization allows us to efficiently apply the algorithm to text mining.

Although our problem setting is that of text mining and NLP, the techniques presented in this paper need not be limited to those domains. We recommend using our approach for information retrieval tasks dealing with sequential similarity when the input data set is too small to be sufficient for machine learning.

3.2.1 OpBerg

The proposed approach, named *OpBerg*, builds upon the AGE [AG11] algorithm: it uses a similar strategy to find the optimal number of local alignments. AGE can be thought of as splitting the input sequences into segments and then running a local alignment algorithm on those segments. The original form of AGE that involves going forward and reverse in two matrices makes any additional alignment gaps difficult to compute and store. It is thus the linear-space algorithm that holds the key to solving the problem of optimal local alignments. Because the directionality moves from left to right (or right to left), this approach can be used to split the strings into an arbitrary number of segments. Further information is needed to implement the proposed approaches that retain necessary information about the locations of the gaps in the alignments. The change required to the original AGE equation is the addition of a matrix that stores the location of a newly created alignment (for brevity we show only the relevant additions to Equation 2.5–Equation 2.8):

$$X(i,j) = \begin{cases} X(i\text{-}1,j), & \text{if } R(i,j) = R(i\text{-}1,j) + Q \\ X(i\text{-}1,j\text{-}1), & \text{if } R(i,j) = R(i\text{-}1,j\text{-}1) + S(a_i,b_j) \\ X(i,j\text{-}1), & \text{if } R(i,j) = R(i,j\text{-}1) + Q \\ (i\text{-}1,j\text{-}1), & \text{if } R(i,j) = M(i,j\text{-}1) + S(a_i,b_j) \\ (0,0), & \text{if } R(i,j) = 0 \end{cases} \tag{3.2}$$

This optimal solution also uses our proposed concept of score length, whose definition is as follows:

Definition: score length. The score length for the alignment of POS tokens $a_i a_{i+1} \ldots a_{i+d_1}$ and $b_j b_{j+1} \ldots b_{j+d_2}$ is defined as the difference between the max score in the alignment matrix at cell locations $(i + d_1,j + d_2)$ and (i,j). As an example, the score length between "BA" and "AM" in Section 2.1.2.1 is (-2) $-$ (-3) $= 1$

A naive algorithm for solving the optimal alignment problem is to run the existing AGE method on every possible number of local alignments that could reasonably occur:

$$L(i,j,0) = Max \left\{ \begin{array}{l} L(i-1,j,0) + Q \\ L(i-1,j-1,0) + S(a_i,b_j) \\ L(i,j-1,0) + Q \\ 0 \end{array} \right\} \tag{3.3}$$

$$L(i,j,k) = Max \left\{ \begin{array}{l} L(i-1,j,k) + Q \\ L(i-1,j-1,k) + S(a_i,b_j) \\ L(i,j-1,k) + Q \\ M(i-1,j-1,k) + S(a_i,b_j) \\ 0 \end{array} \right\} \tag{3.4}$$

$$M(i,j,0) = Max \left\{ \begin{array}{l} M(i-1,j,0) \\ L(i,j,0) \\ M(i,j-1,0) \end{array} \right\} \tag{3.5}$$

$$M(i,j,k) = Max \left\{ \begin{array}{l} M(i-1,j,k) \\ L(i,j,k-1) \\ M(i,j-1,k) \end{array} \right\} \tag{3.6}$$

$$X_I(i{,}j) = X(i-1{,}j{,}k) \tag{3.7}$$

$$X_M(i{,}j) = X(i-1,j-1{,}k) \tag{3.8}$$

$$X_D(i{,}j) = X(i{,}j-1{,}k) \tag{3.9}$$

$$X_X(i{,}j) = X(i{,}j{,}k-1) \cup (i-1,j-1) \tag{3.10}$$

$$X_0 = (0,0) \tag{3.11}$$

$$L_I(i{,}j{,}k) = L(i-1,j,k) + Q \tag{3.12}$$

$$L_M(i{,}j{,}k) = L(i-1,j-1{,}k) + S(a_i,b_j) \tag{3.13}$$

$$L_D(i{,}j{,}k) = L(i,j-1{,}k) + Q \tag{3.14}$$

$$L_X(i{,}j{,}k) = M(i-1,j-1,k) + S(a_i,b_j) \tag{3.15}$$

$$X(i,j,k) = \begin{cases} X_I(i,j), & \text{if } L(i,j,k) = L_I(i,j,k) \\ X_M(i,j), & \text{if } L(i,j,k) = L_M(i,j,k) \\ X_D(i,j), & \text{if } L(i,j,k) = L_D(i,j,k) \\ X_X(i,j), & \text{if } L(i,j,k) = L_X(i,j,k) \\ X_0, & \text{if } L(i,j,k) = 0 \end{cases} \tag{3.16}$$

Although this may seem to be an unreasonable solution, the running time and memory usage remain polynomial and thus feasible for small input sizes.

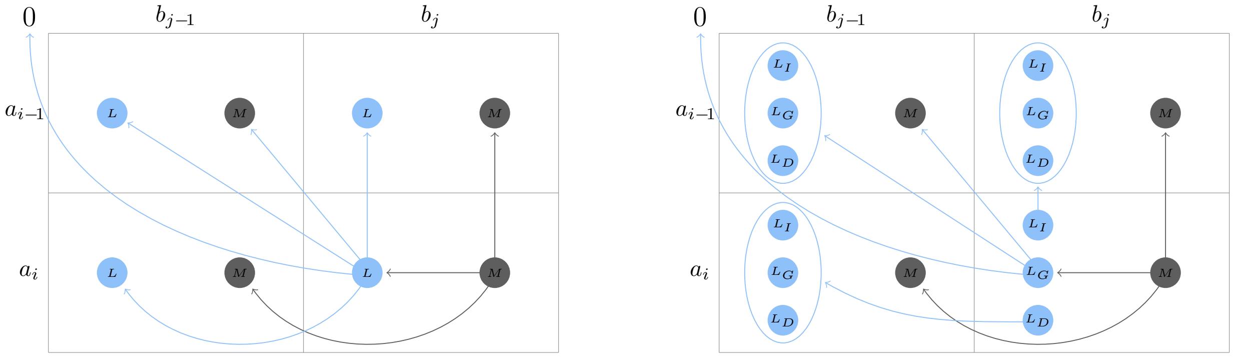
As shown by Equation 3.3–Equation 3.16, the change required is to compute and store the possible different alignments using a separate matrix for each split. A new variable is introduced, k, which represents the current number of local alignments to run on the given input sequences. The results of these additions require an n factor increase in both running time and memory retention, where n is defined as the size of the largest input POS token sequence. The running time becomes $\mathcal{O}(n^3)$ with memory required as $\mathcal{O}(n^3)$.

Like the segmented least squares problem [Bel61], it is intuitive to add a penalty (P) for each additional increase in local alignments. This penalty is needed since otherwise, the optimal alignment would always just match individual POS tokens. Because this penalty is proportional to the number of local alignments, we make the penalty a simple linear constant. The maximum alignment score can then be defined as:

$$\underset{1 \leq k \leq n}{Max}[P \times k + M(|A|, |B|, k)] \tag{3.17}$$

where A and B are the input POS token sequences mapped from two sentences. M is the three-dimensional maximum matrix which holds the maximum alignment score for each a_i, b_j, and k; where $a_i \in A$ and $b_j \in B$.

A simple linear penalty constant reveals that returning one such alignment is not a trivial and deterministic task. The linear penalty can be thought of as an additional larger gap penalty, thus taking the form of a generalized global alignment [HC03]. It has already been shown [AG11] that this can lead to improper alignments.



(a) Linear Gap OpBerg

(b) Affine Gap OpBerg

Figure 3.1: Dependent states to determine the states in the (i,j) cell for OpBerg with a linear gap (a), and affine gap (b).

The question then becomes: What is the optimal number of alignments? For example, a user may prefer to find an alignment that has only 1 large segment aligned and a score of 28 over 10 alignments and a score of 29. To determine the correct number of alignments, this work focuses on three major trade-offs:

1. Number of alignments.
2. Score length to break apart an alignment (α).
3. Minimum score length to start an alignment (β).

The naive algorithm solves the problem of finding the optimal number of local alignments, but it does so at a considerable cost. For causal sentences, this increase is not infeasible due to the relatively low input size of sentences. But running this algorithm over a very large corpus like the entirety of PubMed Central[1] would carry a considerable execution cost. Thus, it is advantageous to seek solutions that are more efficient in both time and space. Opberg, the approach we present here, seeks to reduce memory by a factor of n^2 and execution time by a factor of n^2.

3.2.1.1 Model

Note that during execution of the naive algorithm described above, once it is decided that a new local alignment is a better choice, the optimal solution can then only be of the same or more alignments.

[1] https://www.ncbi.nlm.nih.gov/pmc/

This allows us to reuse the existing M matrix and shave off the k dimension, allowing for much simpler bookkeeping. We introduce a new matrix L that represents the values of a local alignment. The M matrix then takes on the interpretation of a matrix whose values are the max of the previous max M cell value and the corresponding L cell value. The optimal solution then can be in the L matrix (that is, performing a local alignment) or in the M matrix (that is, moving through the cells of the matrix and not decreasing in value). We use the notation that if the optimal solution is in the L matrix, then it is in the "L" or "alignment" state; and if the optimal solution is in the M matrix, then it is in the "M" or "max" state. Given that there is only one L state, it is entirely possible for the optimal solution to transition multiple times from the M state to the L state before beginning an alignment. We store the values of a transition in a new matrix N which holds the point of a transition in and out of the M state. Another matrix X holds the points of all transitions through the optimal solution.

The three trade-offs discussed above can be dealt with in various ways. To account for the number of alignments, we can leave in the original penalty P, but instead of considering this as a larger gap penalty, one can think of it as a value less than 1 and possibly even 0 (with the original gap penalty greater than 1). By doing so, one can easily gauge at what point a new alignment gap starts to weigh negatively on the score and thus becomes less desirable.

To consider the minimum score length that is considered to break apart an alignment, we need only consider the point at which the algorithm exits the max state. If the current alignment has not dropped below the input score length α, then we will restrict the transition until the appropriate threshold has been reached.

Likewise for the start of an alignment, with the change only to the entering of the max state. This requires storing the score at the start of entering the alignment state so that we can compare the difference to see if we are above threshold. This value is stored in the matrix H. This allows us to restrict the length as we do for breaking apart an alignment, but a key difference happens when an alternative alignment is nonexistent. For example, a user may prefer not to start a segment of only 3 matched characters unless this is the max score out of any alternative alignments by a score of 3 matches. We must introduce into this restriction of a transition into the max state a way to keep track of how a score length smaller than β influences the score. That is, we do not necessarily want to discard these alignments unless there is a better alignment available. A new parameter is introduced, $\gamma(x)$, which allows the user to specify a function to weigh how important a certain score length is when it is below threshold, but no higher scoring alternatives exist.

With these parameters, the algorithm is bound to a running time of $\mathcal{O}(n^2)$ and memory requirements of $\mathcal{O}(n^3)$. The intuition for this algorithm follows the intuition of segmented least squares. In the segmented least squares problem, we are searching for a balance between accuracy and number of lines, whereas in OpBerg we seek this parsimony between alignment score and number of jumps through the matrix to start a new local alignment. The trade-off is then enforced by the penalty constants P, α, β, and function $\gamma(x)$.

3.2.1.2 Concurrent Processing

Even with the reduction in the running time as given by the above algorithm it is still desirable to reduce the speed of computation for large input sizes. To reduce the speed of computation, one may search for and take the path of large diagonals in the matrix [PL88, Rog01] or use approximations [Alt90]. The approximations are unusable where exact matching is required, and the large diagonals speedup is dependent on having the right type of input to be effective. We present a simple multi-threaded approach that guarantees the exactness of the result while providing the same speed-up regardless of the inputs.

The approach is to utilize the necessary knowledge to compute a given comparison (cell in the recurrent matrix). As shown by Figure 3.1 all that is necessary for calculation of the given state is 3 previous states. With this restriction we can have a pool of threads that calculate their state given the known information and then continue. The only requirement is that the previous 3 states are ready for the calculation of the current state. With the necessary knowledge being the cell above, diagonal, and left of the current cell of interest, one thread will calculate a cell based on these given states, then the current calculated cell can be passed along to a parallel process which then becomes the necessary left state.

Under this approach the execution time becomes $\mathcal{O}(n^2/T)$ where T is the number of parallel units.

3.2.1.3 Quadratic Space

With the introduction of the concurrent processing approach it is not entirely clear whether the algorithm can be used under existing linear space approaches which utilize a column vector and a divide-and-conquer approach to reduce the memory. We find the existing approach to be easily adaptable to Hirschberg-based methods [Hir75] by maintaining a column vector and shifting the cells as necessary. An additional change is needed that holds the current locations of each cell and restricts updating a cell until the locations of the subsequent thread have moved on from the current cell of interest. Algorithm 3.2.1 details the necessary changes which reduce the memory to $\mathcal{O}(n^2)$.

The existing memory requirements can be reduced even further by trading space for computation time. By limiting the number of items held in the X and N matrices, the memory will be capped by an input number Y. With one run of the algorithm the input can be reduced by the returned Y breakpoints and recursively run. If the optimal number of breakpoints is R then the execution time becomes $\mathcal{O}(\frac{R \times n^2}{Y \times T})$ and memory at $\mathcal{O}(Y \times n)$.

3.2.1.4 Affine Gap

It should not always be the case that insertions and deletions (indels) between the inputs are weighed equally, regardless of where they occur. For instance, in certain causal sentences, a large cluster of indels may represent a tangential segment of words. To capture these occurrences, an affine gap model that takes into account segments of tangential words must be adapted to OpBerg.

The changes required of OpBerg for an affine gap are similar to those in the original local alignment algorithm [Got82]. Three matrices—representing a match/mismatch (L_G), insertion (L_I), and deletion (L_D) transitions, respectively—must be used in place of the original L matrix. The max matrix M cannot enter into any of these three states because it represents a jump through the inputs, so it remains the same. Also, since a local alignment must start and end with a match (diagonal move), the transition between the L states to the M states can occur only through the new L_G matrix. This also applies to the X and N matrices, as they only must monitor jumps between the L_G and the M matrices.

The recurrent relations needed for the affine gap OpBerg model are given in their entirety as:

$$L_I(i,j) = Max \left\{ \begin{array}{l} L_I(i-1,j) + E \\ L_G(i-1,j) + O + E \\ L_D(i-1,j) + O + E \end{array} \right\} \tag{3.18}$$

$$H_I(i,j) = \begin{cases} H_I(i-1,j) & \text{if } L_I(i,j) = L_I(i-1,j) + E \\ H_G(i-1,j) & \text{if } L_I(i,j) = L_G(i-1,j) + O + E \\ H_D(i-1,j) & \text{if } L_I(i,j) = L_D(i-1,j) + O + E \end{cases} \tag{3.19}$$

$$\theta(i,j) = Max \left\{ \begin{array}{l} M(i-1,j) \\ M(i,j-1) \end{array} \right\} \tag{3.20}$$

$$\delta(i,j) = Max \left\{ \begin{array}{l} 0 \\ L_I(i-1,j-1) + S(a_i,b_j) \\ L_G(i-1,j-1) + S(a_i,b_j) \\ L_D(i-1,j-1) + S(a_i,b_j) \end{array} \right\} \tag{3.21}$$

$$L_{G,I,H}(i,j) = L_I(i-1,j-1) + S(a_i,b_j) \tag{3.22}$$

$$L_{G,G,H}(i,j) = L_G(i-1,j-1) + S(a_i,b_j) \tag{3.23}$$

$$L_{G,D,H}(i,j) = L_D(i-1,j-1) + S(a_i,b_j) \tag{3.24}$$

$$L_{G,M,H}(i,j) = M(i-1,j-1) + S(a_i,b_j) + P \tag{3.25}$$

$$\psi(i,j) = \begin{cases} \theta(i,j) & \text{if } \delta(i,j) = 0 \\ H_I(i-1,j-1) & \text{if } \delta(i,j) = L_{G,I,H}(i,j) \\ H_G(i-1,j-1) & \text{if } \delta(i,j) = L_{G,G,H}(i,j) \\ H_D(i-1,j-1) & \text{if } \delta(i,j) = L_{G,D,H}(i,j) \end{cases} \tag{3.26}$$

$$\pi(i,j) = M(i-1,j-1) + S(a_i,b_j) + P \tag{3.27}$$

$$\epsilon(i,j) = \begin{cases} \pi(i,j) & \text{if } \delta(i,j) - \psi(i,j) \leq \alpha \\ -\infty & \text{otherwise} \end{cases} \tag{3.28}$$

$$L_G(i,j) = Max \left\{ \begin{array}{c} \delta(i,j) \\ \epsilon(i,j) \end{array} \right\} \tag{3.29}$$

$$H_G(i,j) = \begin{cases} \theta(i,j) & \text{if } L_G(i,j) = 0 \\ H_I(i-1,j-1) & \text{if } L_G(i,j) = L_{G,I,H}(i,j) \\ H_G(i-1,j-1) & \text{if } L_G(i,j) = L_{G,G,H}(i,j) \\ H_D(i-1,j-1) & \text{if } L_G(i,j) = L_{G,D,H}(i,j) \\ \theta(i,j) & \text{if } L_G(i,j) = L_{G,M}(i,j) \end{cases} \tag{3.30}$$

$$L_D(i,j) = Max \left\{ \begin{array}{c} L_I(i,j-1) + O + E \\ L_G(i,j-1) + O + E \\ L_D(i,j-1) + E \end{array} \right\} \tag{3.31}$$

$$L_{D,I,H}(i,j) = L_I(i,j-1) + O + E \tag{3.32}$$

$$L_{D,G,H}(i,j) = L_G(i,j-1) + O + E \tag{3.33}$$

$$L_{D,D,H}(i,j) = L_D(i,j-1) + E \tag{3.34}$$

$$H_D(i,j) = \begin{cases} H_I(i,j-1) & \text{if } L_D(i,j) = L_{D,I,H}(i,j) \\ H_G(i,j-1) & \text{if } L_D(i,j) = L_{D,G,H}(i,j) \\ H_D(i,j-1) & \text{if } L_D(i,j) = L_{D,D,H}(i,j) \end{cases} \tag{3.35}$$

$$\zeta(i,j) = \begin{cases} L_G(i,j) & \text{if } L_G(i,j) \geq \beta \\ \gamma(L_G(i,j)) & \text{otherwise} \end{cases} \tag{3.36}$$

$$M(i,j) = Max \left\{ \begin{array}{l} \zeta(i{,}j) \\ M(i-1,j) \\ M(i,j-1) \end{array} \right\} \tag{3.37}$$

$$L_{G,I,X}(i,j) = L_I(i-1,j) + Q \tag{3.38}$$

$$L_{G,G,X}(i,j) = L_G(i-1,j-1) + S(a_i{,}b_j) \tag{3.39}$$

$$L_{G,D,X}(i,j) = L_D(i,j-1) + Q \tag{3.40}$$

$$N_X(i,j) = N(i-1,j-1) \cup (i,j) \tag{3.41}$$

$$X_D(i,j) = X(i-1,j-1) \tag{3.42}$$

$$X(i,j) = \begin{cases} X(i-1,j) & \text{if } L_G(i,j) = L_{G,I,X}(i,j) \\ X_D(i,j) & \text{if } L_G(i,j) = L_{G,G,X}(i,j) \\ X(i,j-1) & \text{if } L_G(i,j-1) = L_{G,D,X}(i,j) \\ N_X(i,j) & \text{if } L_G(i,j) = \epsilon(i{,}j) \\ \varnothing & \text{if } L_G(i,j) = 0 \end{cases} \tag{3.43}$$

$$N(i,j) = \begin{cases} X(i,j) \cup (i,j), & \text{if } M(i,j) = \zeta(i{,}j) \\ N(i-1,j), & \text{if } M(i,j) = M(i-1,j) \\ N(i,j-1), & \text{if } M(i,j) = M(i,j-1) \end{cases} \tag{3.44}$$

where $(i{,}j)$ represents the cell location of both matrices and the ith POS token in A and the jth POS token in B. S is a function that takes in two POS tokens and returns a score value. The opening gap penalty is represented by O and the extension penalty by E.

Even with the newly created matrices and additional processing that must take place to populate the matrices, the running time will be $\mathcal{O}(n^2)$, with memory as $\mathcal{O}(n^2)$.

3.2.1.5 Path Reconstruction

The existing techniques presented so far have neglected to discover the path used to find the maximal local alignments. The return values have only consisted of the maximum score and a set of breakpoints derived from transitions from the M to the L_G matrices. From these breakpoints the path can be easily recovered using traditional techniques [MM88] that find the local alignment path with an affine gap. The inputs can be broken into substrings along the breakpoints and then fed into a subroutine calculating the normal local alignment with an affine gap. The results will be matched with the substring and returned. Even though the traditional techniques are not concurrent, the existing concurrency method is also applicable to these approaches. Therefore, the existing execution time and memory size does not change.

3.2.1.6 Linear Space

Hirschberg-based methods have already shown the ability to linearize existing quadratic space algorithms. It is a reasonable approach to apply this technique to the optimal alignments problem. In fact, the solution does not differ very much from the techniques that have previously been applied. The difference between OpBerg and other classical alignment problems dealing with affine gap is the addition of the jump (M matrix) state. Outside of this state, the problem is the same as the affine gap solution with the only change being higher starting values.

The solution to the linear affine gap problem is to consider the fact that the point on an optimal path must be in one of two different states. These states are the deletion state or the diagonal state (assuming indel penalties are less than match/mismatch penalties). The reason the maximum point in the column vector that bisects the matrix cannot be in the insertion state is due to the fact that this would require another point in the same column vector to be a higher value. With the addition of the jump state, we consider whether the optimal path is in this jump state–much like the consideration of whether the optimal path is in a deletion or diagonal state. Since the transition between the jump state to an alignment state must occur through a diagonal, we can just consider this case as a normal diagonal move.

Algorithm 3.2.1 Affine OpBerg

Input: Sequences a and b, new alignment penalty P, gap opening penalty O, gap extending penalty E, scoring function S.
Output: Maximum score, set of breakpoints

```
procedure SCORE($a$, $b$)
    if $|a| > |b|$ then
        $swap(a, b)$
    end if
    for $j = 1$ to 2 do
        for $i = 1$ to $|a|$ do
            $L_I(i, j) \leftarrow -\infty$
            $L_G(i, j) \leftarrow 0$
            $L_D(i, j) \leftarrow -\infty$
            $M(i, j) \leftarrow 0$
            $X(i, j) \leftarrow \{\}$
            $N(i, j) \leftarrow \{\}$
            $A(i, j) \leftarrow (i,j)$
        end for
    end for
    $\alpha \leftarrow -\infty$
    $\mu \leftarrow$ (0,0)
    for $j = 1$ to $|b|$ in parallel do
        for $i = 1$ to $|a|$ do
            $c \leftarrow$ true
            while $c =$ true do
                $c \leftarrow A(i-1, j) \neq (i-1, j)$                  (a)
                $c \leftarrow c \vee A(i-1, j-1) \neq (i-1, j-1)$
                $c \leftarrow c \vee A(i, j-1) \neq (i, j-1)$
            end while
            Calculate $L_I(i, j)$ according to Equation 3.18
            Calculate $L_G(i, j)$ according to Equation 3.29
            Calculate $L_D(i, j)$ according to Equation 3.31
            Calculate $M(i, j)$ according to Equation 3.37
            Calculate $X(i, j)$ according to Equation 3.43
            Calculate $N(i, j)$ according to Equation 3.44
            if $M(i, 1) > \alpha$ then
                lock $\alpha$
                    if $M(i, 1) > \alpha$ then
                        $\alpha \leftarrow M(i, 1)$
                        $\mu \leftarrow X(i,1)$
                    end if
                end lock
            end if
            $L_I(i-1, 1) \leftarrow L_I(i-1, 2)$
            $L_G(i-1, 1) \leftarrow L_G(i-1, 2)$
            $L_D(i-1, 1) \leftarrow L_D(i-1, 2)$
            $M(i-1, 1) \leftarrow M(i-1, 2)$
            $X(i-1, 1) \leftarrow X(i-1, 2)$
            $N(i-1, 1) \leftarrow N(i-1, 2)$
            if $j = m$ then                                      (b)
                $L_I(i, 1) \leftarrow L_I(i, 2)$
                $L_G(i, 1) \leftarrow L_G(i, 2)$
                $L_D(i, 1) \leftarrow L_D(i, 2)$
                $M(i, 1) \leftarrow M(i, 2)$
                $X(i, 1) \leftarrow X(i, 2)$
                $N(i, 1) \leftarrow N(i, 2)$
            end if
            $A(i-1, 1) \leftarrow (i-1, j)$
            $A(i-1, 2) \leftarrow (i-1, j+1)$
            if $j = m$ then                                      (c)
                $A(i, 1) \leftarrow (i, j)$
                $A(i, 2) \leftarrow (i, j+1)$
            end if
        end for
    end for
    return $\alpha, \mu$
end procedure
```

The forward recurrences become:

$$I_f(i,j) = Max \left\{ \begin{array}{l} I_f(i-1,j) + E \\ G_f(i-1,j) + O + E \\ D_f(i-1,j) + O + E \end{array} \right\} \tag{3.45}$$

$$\delta(i,j) = Max \left\{ \begin{array}{l} I_f(i-1,j-1) + S(a_i,b_j) \\ G_f(i-1,j-1) + S(a_i,b_j) \\ D_f(i-1,j-1) + S(a_i,b_j) \\ 0 \end{array} \right\} \tag{3.46}$$

$$\pi(i{,}j) = M_f(i-1,j-1) + S(a_i,b_j) + P \tag{3.47}$$

$$\epsilon(i{,}j) = \begin{cases} \pi(i{,}j) & \text{if } \delta(i,j) - M_f(i,j) \leq \alpha \\ -\infty & \text{otherwise} \end{cases} \tag{3.48}$$

$$G_f(i,j) = \begin{cases} \epsilon(i{,}j) & \text{if } \epsilon > \delta(i,j) \\ \pi(i,j) & \text{otherwise} \end{cases} \tag{3.49}$$

$$D_f(i,j) = Max \left\{ \begin{array}{l} I_f(i,j-1) + O + E \\ G_f(i,j-1) + O + E \\ D_f(i,j-1) + E \end{array} \right\} \tag{3.50}$$

$$\zeta(i{,}j) = \begin{cases} G_f(i,j) & \text{if } G_f(i,j) \geq \beta \\ \gamma(G_f(i,j)) & \text{otherwise} \end{cases} \tag{3.51}$$

$$M_f(i,j) = Max \left\{ \begin{array}{l} M_f(i-1,j) \\ \zeta(i{,}j) \\ M_f(i,j-1) \end{array} \right\} \tag{3.52}$$

With the backwards recurrences as:

$$I_b(i,j) = Max \begin{Bmatrix} I_b(i+1,j) + E \\ G_b(i+1,j) + O + E \\ D_b(i+1,j) + O + E \end{Bmatrix} \tag{3.53}$$

$$\delta(i,j) = Max \begin{Bmatrix} I_b(i+1,j+1) + S(a_i,b_j) \\ G_b(i+1,j+1) + S(a_i,b_j) \\ D_b(i+1,j+1) + S(a_i,b_j) \\ 0 \end{Bmatrix} \tag{3.54}$$

$$\pi(i,j) = M_b(i+1,j+1) + S(a_i,b_j) + P \tag{3.55}$$

$$\epsilon(i,j) = \begin{cases} \pi(i,j) & \text{if } \delta(i,j) - M_b(i,j) \leq \alpha \\ -\infty & \text{otherwise} \end{cases} \tag{3.56}$$

$$G_b(i,j) = \begin{cases} \epsilon(i,j) & \text{if } \epsilon(i,j) > \delta(i,j) \\ \delta(i,j) & \text{otherwise} \end{cases} \tag{3.57}$$

$$D_b(i,j) = Max \begin{Bmatrix} I_b(i,j+1) + O + E \\ G_b(i,j+1) + O + E \\ D_b(i,j+1) + E \end{Bmatrix} \tag{3.58}$$

$$\zeta(i,j) = \begin{cases} G_b(i,j) & \text{if } G_b(i,j) \geq \beta \\ \gamma(G_b(i,j)) & \text{otherwise} \end{cases} \tag{3.59}$$

$$M_b(i,j) = Max \begin{Bmatrix} M_b(i+1,j) \\ \zeta(i,j) \\ M_b(i,j+1) \end{Bmatrix} \tag{3.60}$$

With the affine gap the solution must always remember whether the first forward or backward move is in a deletion state. In these cases, the subsequent deletion moves will be applied as a linear case instead of opening a gap. To determine the state which is maximal in the column vector bisecting the matrix, variables holding an opening delete (o), extension delete (e), jump (m), match/mismatch (d), transition from a jump state (t_f), transition to a jump state (t_b), and transition to and from a jump state (t) state must be evaluated under the following equations:

$$o_I(i,j) = I_f(i,j-1) + I_b(i+1,j+1) + O + E \tag{3.61}$$

$$o_G(i,j) = G_f(i,j-1) + G_b(i+1,j+1) + O + E \tag{3.62}$$

$$o(i,j) = Max \left\{ \begin{array}{c} o_I(i,j) \\ o_G(i,j) \end{array} \right\} \tag{3.63}$$

$$e(i,j) = D_f(i,j-1) + D_b(i+1,j+1) + E \tag{3.64}$$

$$m(i,j) = M_f(i,j-1) + M_b(i+1,j+1) \tag{3.65}$$

$$d_f(i,j) = Max \left\{ \begin{array}{c} I_f(i-1,j-1) + S(a_i,b_j) \\ G_f(i-1,j-1) + S(a_i,b_j) \\ D_f(i-1,j-1) + S(a_i,b_j) \end{array} \right\} \tag{3.66}$$

$$d_b(i,j) = Max \left\{ \begin{array}{c} I_b(i+1,j+1) + S(a_i,b_j) \\ G_b(i+1,j+1) + S(a_i,b_j) \\ D_b(i+1,j+1) + S(a_i,b_j) \end{array} \right\} \tag{3.67}$$

$$d(i,j) = d_f(i,j) + d_b(i,j) - S(a_i,b_j) \tag{3.68}$$

$$t_f(i,j) = M_f(i-1,j-1) + P + d_f(i,j) \tag{3.69}$$

$$t_b(i,j) = M_b(i+1,j+1) + P + d_b(i,j) \tag{3.70}$$

$$t(i,j) = M_f(i\text{-}1,j\text{-}1) + M_b(i\text{+}1,j\text{+}1) + P + S(a_i,b_j) \tag{3.71}$$

The above equations handle the correct score in the majority of cases. A few edge cases occur given the requirements of P, α, β and $\gamma(x)$. It may be the case that P is added an additional time, if both the forward and backward alignments have transitioned out for the matrix state. For this we just need to add an additional boolean matrix that records whether there was a transition. If a transition did occur, we just add a P back to the score where the current state is in an alignment.

Algorithm 3.2.2 OpBerg

Input: Sequences a and b, new alignment penalty P, gap opening penalty O, gap extending penalty E, scoring function S.
Output: Maximum score, alignment path

```
procedure SCORE($a$, $b$, $P$, $O$, $E$)
    $z \leftarrow 0$
    if $|a| > |b|$ then
        $swap(a, b)$
    end if
    return $Linear(a, b, O, z, O, z, O, z, O, z, P, O, E)$
end procedure

procedure LINEAR($a$, $b$, $i_f$, $g_f$, $d_f$, $m_f$, $i_b$, $g_b$, $d_b$, $m_b$, $P$, $O$, $E$)
    Initialize $I$, $G$, $D$ $M$ from to $i$, $g$, $d$, $m$
    for $i = 1$ to $|a|$ do
        Calculate $I_f(i, 0)$ according to Equation 3.45
        $G_f(i, 0) \leftarrow -\infty$
        $D_f(i, 0) \leftarrow -\infty$
        $M_f(i, 0) \leftarrow m_f$
        $A_f(i, 0) \leftarrow (i, 0)$
    end for
    Calculate $D_f(0, 1)$ according to Equation 3.50
    for $i = |a| - 1$ to 1 do
        Calculate $I_b(i, 1)$ according to Equation 3.53
        $G_b(i, 1) \leftarrow -\infty$
        $D_b(i, 1) \leftarrow -\infty$
        $M_b(i, 1) \leftarrow m_b$
        $A_b(i, 1) \leftarrow (i, 1)$
    end for
    Calculate $D_b(|a|, 1)$ according to Equation 3.58
    for $j = 1$ to $\lceil |b|/2 \rceil - 1$ in parallel do
        for $i = 1$ to $|a|$ do
            Wait for $A_f$ according to Algorithm 3.2.1(a)
            Calculate $I_f$, $G_f$, $D_f$, $M_f$ according to Equations 3.45–3.52
            Shift $I_f$, $G_f$, $D_f$, $M_f$ according to Algorithm 3.2.1(b)
            Set $A_f$ according to Algorithm 3.2.1(c)
        end for
    end for
    for $j = |b|$ to $\lceil |b|/2 \rceil + 1$ in parallel do
        for $i = |a|$ to 1 do
            Wait for $A_b$ according to Algorithm 3.2.1(a) (rev)
            Calculate $I_b$, $G_b$, $D_b$, $M_b$ according to Equations 3.53–3.60
            Shift $I_b$, $G_b$, $D_b$, $M_b$ according to Algorithm 3.2.1(b) (rev)
            Set $A_b$ according to Algorithm 3.2.1(c) (rev)
        end for
    end for
    $\alpha \leftarrow -\infty$
    $s \leftarrow 0$
    $h \leftarrow \lceil |b|/2 \rceil$
    for $i = 1$ to $|a|$ do
        Calculate $o, e, m, d, t_f, t_b, t$ according to Equations 3.61–3.71
        $max \leftarrow Max(o, e, m, d, t_f, t_b, t)$
        if $max > \alpha$ then
            $\alpha \leftarrow max$
            $s \leftarrow i$
            Set $i_f$, $i_b$, $g$, $d$, $m$ according to move
        end if
    end for
    $a_l \leftarrow a_{1,2,\ldots,s-1}$
    $a_r \leftarrow a_{s+1,s+2,\ldots,|a|}$
    $b_l \leftarrow b_{1,2,\ldots,h-1}$
    $b_r \leftarrow b_{h+1,h+2,\ldots,|b|}$
    $left \leftarrow Linear(a_l, b_l, i_f, g_f, d_f, m_f, i_b, g_b, d_b, m_b)$
    $right \leftarrow Linear(a_r, b_r, i_f, g_f, d_f, m_f, i_b, g_b, d_b, m_b)$
    return $left \cup \{\alpha, (s, h)\} \cup right$
end procedure
```

The other edge case to consider is when the backward and forward alignments are both in a state of an alignment that is less than β. It is possible that the sum may be greater than β or even a separate call to $\gamma(x)$ must be made. This only requires a call to the original M equation but with the consideration of both the forward and backward alignments for the case when the alignments are in a small score length alignment state.

The linear approach also has the benefit of discarding the N and X matrices. The breakpoints will no longer need to be recorded as we go through the matrix since they will be discovered linearly in a divide-and-conquer manner through the matrix. The existing multithreaded algorithms apply in the same way as in the linear space version. Algorithm 3.2.1 shows the complete approach which results in a solution that costs $\mathcal{O}(n^2/T)$ in execution time and $\mathcal{O}(n)$ in memory.

3.2.2 Evaluation

To show the utility of OpBerg, we demonstrate its ability to find causal sentences and compare it to other models. We also run a set of empirical tests, which shows OpBerg to be well suited for various tasks. After investigating the performance of our algorithm, we describe in more detail a causal example, followed by experiments showing OpBerg's ability to identify causal sentences.

3.2.2.1 Performance Benchmarking

To show the utility of the concurrent processing approach, we set up a simple experiment where we compare two sequences of size n using 1 to 3 threads. The input size is varied from 100 to 10,000 characters. The results which are plotted in Figure 3.2, show that the execution does indeed follow a $\mathcal{O}(n^2)$ function, but adding in additional threads decreases the execution time. At an input size of 10,000, the processing time for 3 threads is approximately half that of 1 thread. This reduction is likely to be even more significant as the input size grows, which makes the concurrent processing approach useful in large input sets, such as the human genome.

It is often assumed that a quadratic-running-time algorithm is too slow to be feasible for analysis. We show that this is not the case when using OpBerg, compared to using a popular platform such as Weka [HFH09] for analysis. From Weka, we use the four most popular machine learning baseline methods for causality extraction. These are logistic regression (LR), random forest

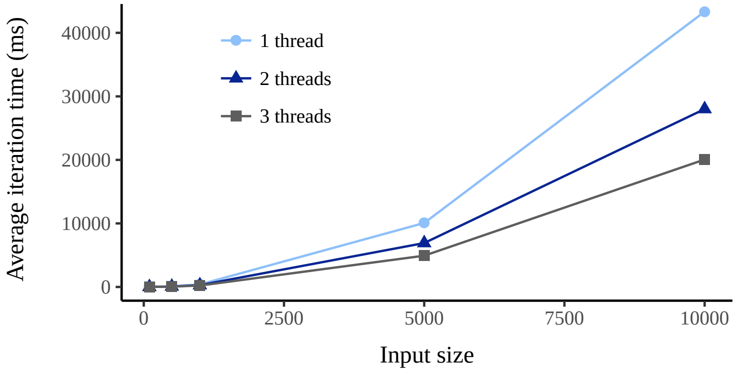


Figure 3.2: Execution time of OpBerg against varying input sizes for 1, 2, and 3 threads.

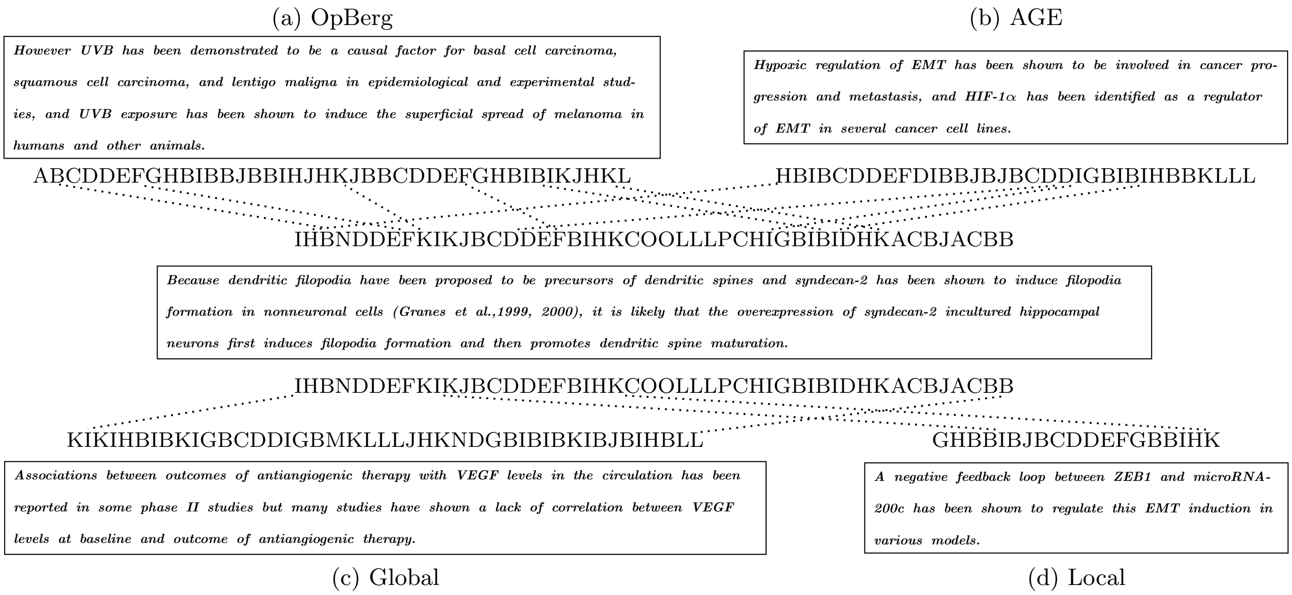
(RF) [LW02], support vector machines (SVM), and naive Bayes classifiers (NBC). We set up a simple experiment where we wish to classify causality of biological sentences with a training set of 100 labeled sentences and a test set that ranges from 1,000 to 10,000 candidate sentences. The results, which are plotted in Table 3.1, show that the execution time of OpBerg far outperforms that of the baseline methods. This is due to the ability to process the input concurrently, which is an advantage of the model. The ability to execute concurrent processing of the baseline methods is dependent on the platform in which they are run, and at least some platforms do not provide concurrency implementations—as is the case with Weka.

Input size	**OpBerg**	**LR**	**RF**	**SVM**	**NBC**
1,000	**248**	495951	14001	12703	44901
5,000	**598**	1013126	72434	61958	416065
10,000	**934**	1496572	138446	122060	867166

Table 3.1: Execution time (ms) of multi-threaded OpBerg alongside logistic regression, random forest, support vector machines, and naive Bayes classifiers for varying input sizes (n) for causality discovery.

3.2.2.2 PubMed Extraction

An integral part of neuroscience initiatives, such as our work in Section 3.3, is cataloging and connecting causal assertions of biomedical findings. To show that OpBerg can automatically discover causal assertions, we give an example of assertions found by OpBerg in a corpus of



(a) OpBerg

However UVB has been demonstrated to be a causal factor for basal cell carcinoma, squamous cell carcinoma, and lentigo maligna in epidemiological and experimental studies, and UVB exposure has been shown to induce the superficial spread of melanoma in humans and other animals.

(b) AGE

Hypoxic regulation of EMT has been shown to be involved in cancer progression and metastasis, and HIF-1α has been identified as a regulator of EMT in several cancer cell lines.

Because dendritic filopodia have been proposed to be precursors of dendritic spines and syndecan-2 has been shown to induce filopodia formation in nonneuronal cells (Granes et al.,1999, 2000), it is likely that the overexpression of syndecan-2 incultured hippocampal neurons first induces filopodia formation and then promotes dendritic spine maturation.

Associations between outcomes of antiangiogenic therapy with VEGF levels in the circulation has been reported in some phase II studies but many studies have shown a lack of correlation between VEGF levels at baseline and outcome of antiangiogenic therapy.

A negative feedback loop between ZEB1 and microRNA-200c has been shown to regulate this EMT induction in various models.

(c) Global

(d) Local

ABCDDEFGHBIBBJBBIHJHKJBBCDDEFGHBIBIKJHKL
HBIBCDDEFDIBBJBJBCDDIGBIBIHBBKLLL
IHBNDDEFKIKJBCDDEFBIHKCOOLLLPCHIGBIBIDHKACBJACBB
IHBNDDEFKIKJBCDDEFBIHKCOOLLLPCHIGBIBIDHKACBJACBB
KIKIHBIBKIGBCDDIGBMKLLLJHKNDGBIBIBKIBJBIHBLL
GHBBIBJBCDDEFGBBIHK

Figure 3.3: A graphical example of matching potential causal sentences using POS mappings. The base sentence [Lin07] was taken from ResearchMaps, found in the PubMed Central corpus, and used as a reference point to find similar sentences under the OpBerg (a) [Shi10], AGE (b) [Vas10], Global (c) [RB10] and Local (d) [Wel10] alignment methods. The dotted lines represent the aligned segments.

biomedical text, as well as those found by other methods. This example is meant to provide the reader with a visual intuition as to why OpBerg works and not meant for large data comparative analysis.

3.2.2.2.1 Experimental Setup

A subset of potential causal sentences was taken from the PubMed Central data set. PubMed Central is an open-access corpus of biomedical literature consisting of over 4 million full-text articles.

Each PubMed Central document is broken into a set of sentences using the Stanford CoreNLP tokenizer. Each sentence is filtered to identify whether it has two or more entities from the Neuroscience Information Framework (NIF) ontology[2]. These candidate sentences are then broken into their respective parts of speech (POS) using the Stanford CoreNLP POS tagger. The NIF filtering was applied to a randomly chosen subset of PubMed Central documents, resulting in 10,000 candidate sentences that originated from 95 distinct research articles.

[2] https://neuinfo.org/

Next, the existing ResearchMaps database was queried to obtain a list of agent–target pairs, as well as the research article in which the agent–target pair appeared. Each article from an agent–target pair was then obtained and split into sentences. The sentences that contain both the agent and targets were taken to be causal sentences and were manually verified. This ground truth consisted of 111 sentences found from a pool of 5,895 experiments in the ResearchMaps database. The sentences were then broken into their respective POS mappings, and the longest sentence was taken as the key in which to map potential causal sentences with.

Each of the 10,000 potential causal sentences from PubMed were aligned against the selected ResearchMap sentence using POS mappings of the sentences. The goal is to find the best matching sentence that captures a causal connection, as well as the type of study that was performed, using the OpBerg, local, global, and AGE algorithms. For each algorithm, a match/mismatch penalty was given as +2/–1, indel as -2 for opening and -1 for an extension. For OpBerg, a new alignment penalty P was set to –1/2, α as 6, β as 3 and $\gamma(x) = 0.5 \times x$. The parameters were chosen out of what we believe is the most traditionally used parameters for these variables [Wik20]. β, α, and $\gamma(x)$ are chosen to what we think is most intuitive. What we think are important are alignments that have some variance (i.e. mismatches and indels) which lead to a high α, and want matches to be longer than single POS matches (β). However, for an additional alignment that gives us a max alignment score we would want to consider it given it is not a single POS match ($\gamma(x)$). This experiment and choice of parameters underscore a larger concept of OpBerg. The algorithm is not meant to be a classifier in the traditional sense, rather it is to find appropriate matches given a small set of input. The only ground truth we have for each input item is the input item itself so any choice of parameters cannot be learned or optimized and must be somewhat arbitrary.

Name	Description	Classes	D	Sentences
RM46	ResearchMaps collection of 46 neuroscience articles	2,4,7	46	200
LLL05	Causal sentences and papers extracted from the LLL05 Challenge	2	45	131
NDE27	Non-domain experts labeling of various PubMed articles	2	27	1,025
BioCause	The biomedical discourse causality corpus and corresponding articles	2	20	1,000
RM6	Domain expert labeled set of articles from ResearchMaps	2,4,7	6	356

Table 3.2: Datasets used in the evaluation of OpBerg in a classification task.

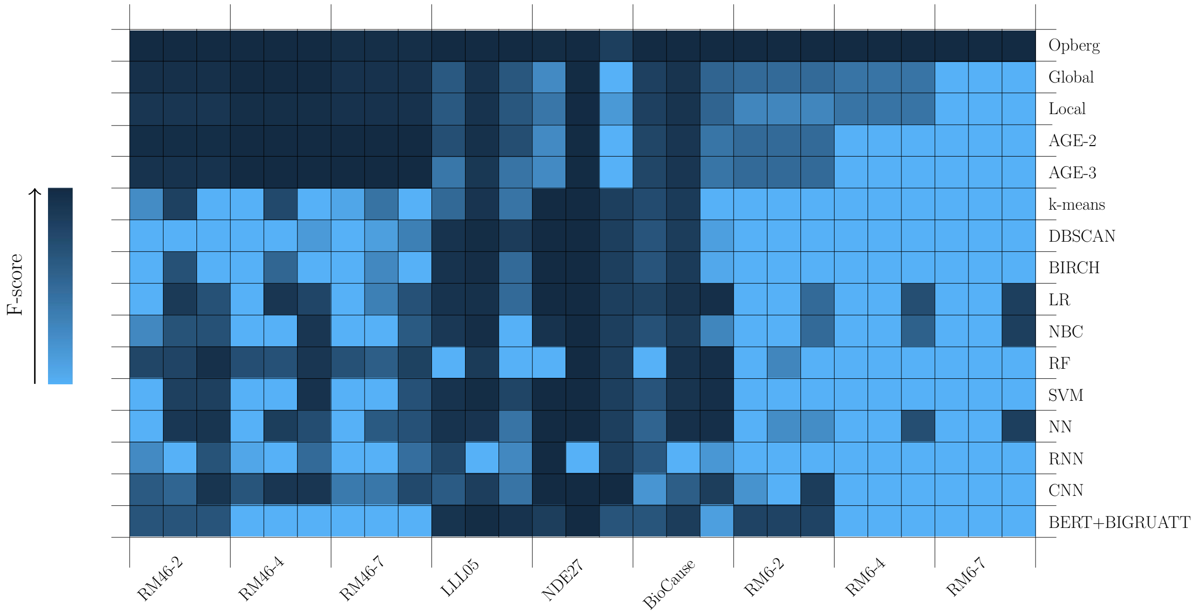


Figure 3.4: F-scores shown as a heatmap for OpBerg compared with baseline methods using POS features of sentences, general word embeddings, and causal embeddings in order to determine causality and other biological classes.

3.2.2.2.2 Experimental Results

The matching sentence for each model is given by Figure 3.3. Out of all the algorithms, OpBerg maintains the highest score (32) over AGE (27), global (7), and local alignments (20). All returned sentences show some form a causality, but to be most useful in research contexts, it is important to identify not only causality but also the experiment type and result. The sentence from AGE indicates a causal connection between HIF-1α and EMT, but no information is given as to how these two entities interact. The sentence used as the key clearly indicates a positive intervention (on syndecan-2) and an excitatory relationship between syndecan-2, filopdia formation, and dendritic spine maturation. This lack of information regarding the study and relationship types also occurs in the global sentence and local sentences. However, the returned OpBerg sentence discovers both the causal relation and the type of study that was performed. The relationship of UVB exposure and melanoma is the exact same as that of the key and therefore is the most useful match among the competing models.

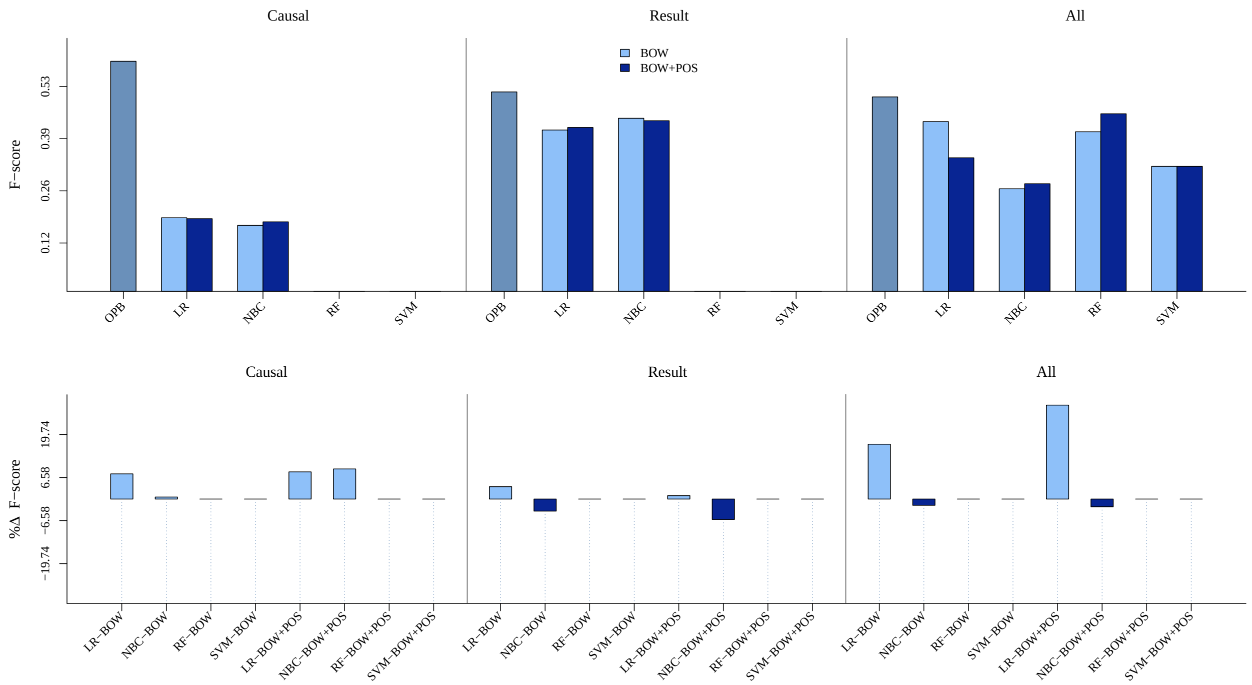


Figure 3.5: F-scores for OpBerg (OPB) compared with LR, NBC, RF, and SVM using a bag of words representation of the input sentences with and without part of speech features in order to determine causality (Causal), the appropriate result class (Result) and all relevant neuroscience classes (All) (a). Results are also shown as a percent change increase in F-score when OpBerg features are included in the baseline models (b).

3.2.2.3 General Classification

A useful feature of OpBerg is its ability to determine whether a sentence is a good match to an existing labeled sentence; it thus captures the similarity between two sentences. This similarity measure then can be used to classify whether the sentence is causal. Furthermore, we seek the ability to classify a more diverse set of classes that are useful to biologists. We test the ability of OpBerg to classify causality against the baseline methods of logistic regression (LR), support vector machines (SVM), naive Bayes classifier (NBC), random forest (RF) [LW02], AGE, the local alignment algorithm (local), global alignment algorithm (global), k-means clustering, density-based spatial clustering of applications with noise (DBSCAN) [Est96], balanced iterative reducing and clustering using hierarchies (BIRCH) [Zha96], a feedforward neural network (NN), and a recurrent neural network (RNN), convolutional neural network (CNN) and lastly a BERT based bidirectional

GRU with self attention (BERT+BIGRUATT) [KAS]. The various datasets used are described by Table 3.2.

3.2.2.3.1 Experimental Setup

For each dataset that contains only causal sentences, we query the PubMed Central corpus for their respective articles. For each article we obtain the sentences which are not like a causal sentence. We define similar as the global alignment score and remove the top 5 most similar sentences to any causal sentence. All remaining sentences are labeled as non-causal. The resultant number of sentences and articles for each dataset is given in Table 3.2.

The hyperparameters for OpBerg were optimized using Bayesian optimization. For each labeled input set we then trained OpBerg, and the baseline methods using 10-fold cross validation. To obtain a classification probability for each alignment method, each test sentence's POS tokens (a) were compared to each POS-mapping of the sentences in the training set using the OpBerg algorithm. The highest matching score (H_m) was taken as the best match for a given sentence's POS string (b). If the best match belonged to a causal sentence, a probability was given to the test item as $P^+(a, H_m)$, with P^+ defined as:

$$P^+(a, H) = \frac{H}{Max[S(a_i, a_i)] \times |a|} \tag{3.72}$$

For each test input we also recorded the highest matching score aligned to a causal training sentence (H_c). If the best match was not a causal sentence, the probability was given to be:

$$P^-(a, H) = Min \left\{ \begin{array}{l} 1 - P^+(a, H_m) \\ \\ P^+(a, H_c) \end{array} \right\} \tag{3.73}$$

Inputs for all methods were made using part of speech (POS) features obtained by the Stanford CoreNLP POS tagger, embeddings using the GloVe Wikipedia 2014 + Gigaword dataset [Pen] and and from a task-specific word embedding technique for causality [BS21].

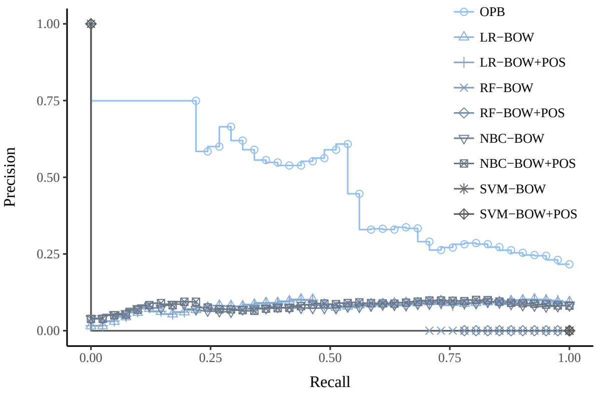


Figure 3.6: The precision recall curve for OpBerg and competing models for the task of determining whether a given sentence is causal.

In some of the datasets we were only able to obtain whether a sentence was causal or not. However, in the ResearchMaps datasets we were able to determine more numerous classes of interest. The classes of interest were separated into three sets. The simplest set was labeled whether a sentence was causal. In the second, we are given the qualitative result of the experiment, i.e., increase, decrease, no change, or non-causal. The most diverse set consisted of 7 different classes describing biological phenomena: a permutation of the set negative or positive with the set of no change, increase, or decrease together with a non-causal label.

3.2.2.3.2 Experimental Results

The F-score for all models and datasets is represented as a heat map in Figure 3.4. In all but one dataset, OpBerg gives the highest F-score amongst each baseline method. Our results suggest alignments are a good technique in classifying causal sentences when the training set size is low; and the best among these alignment algorithms is OpBerg.

3.2.2.4 Domain Expert Classification

3.2.2.4.1 Experimental Setup

From ResearchMaps and the PubMed Central corpus we obtain a ground truth set of sentences as in Section 3.2.2.2. Next, we randomly select a set of sentences that are not causal. The non-causal sentences were first randomly obtained from PubMed Central, along with their POS mappings, and then manually verified as not causal. The result was 81 causal and 119 non-causal sentences. The 81 causal sentences were comprised from a total of 6 research articles.

With this labeled input set we then trained OpBerg, LR, RF, NBC, and SVM by using all but one research article and tested every sentence in the left-out article. We repeat this process for each individual research article. The classification probabilities were taken from Equation 3.72 and Equation 3.73.

Inputs for LR, RF, SVM, and NBC were made using a bag-of-words (BOW) representation, and a combination of BOW and part of speech (POS) features. Probabilities were also calculated for the baseline methods combined with OpBerg features for comparative analysis. The classes of interest are described in Section 3.2.2.3.

3.2.2.4.2 Experimental Results

The F-score for all the models and the classification tasks are given by Figure 3.5(a). Results that were computed using a combination of BOW and POS features are represented in the figure by "BOW+POS". In each task, OpBerg gives the highest F-score compared to each baseline method, regardless of the input. The addition of POS features to the BOW representation appears negligible, but the addition of OpBerg features shows a more profound effect. Figure 3.5(b) demonstrates that more models have a positive change in F-score than models where there is a decrease. And the positive cases show a higher increase in F-score than the negative decreases. This substantiates OpBerg features as useful in existing machine learning methods of causality extraction.

When mining causal connections in text, precision is more important than recall. This is due to having so many candidate items from large document collections such as PubMed Central. Most of the high F-score values for the machine learning models come from a threshold resulting in small precision and large recall. But this is the opposite of what we want: it would result in noisy and incorrect assertions. As is shown in Figure 3.6, only OpBerg displays a precision–recall curve which allows for a threshold to be set that results in a high precision and a high F-score. This ability makes OpBerg ideal in returning a relativity large set of accurate causal classifications.

3.2.3 Discussion

Opberg represents a new approach to solving difficult NLP problems. Opberg is not meant to replace previous state of the art techniques in all datasets, rather it can be the best approach when the classification is complex and only a small amount (in the order of 10^2 to 10^4) labeled data exists. We maintain the effectiveness of Opberg over other machine learning approaches [TJ97, Gir10, Do11, HY10, CC04, Bla08] when the labeled training data is small. Additionally, if confidence is maintained in the quality of the knowledge source in covering the input data, predefined knowledge methods [PB21, Che21b, KB91, GM02, Bui10] may be more appropriate. In situations where an alternative method is preferred, Opberg can add value as an additional learning feature. In fact, it is the vision of the authors that Opberg will be most valuable to the research community as a complement to established approaches.

The benefits of Opberg come at an additional cost. The algorithm is complex, and the context can be unfamiliar to both the trained computer scientist and/or bioinformatician. If execution time is not a concern, it is recommended to implement either Equations 3.3–3.16 or Equations 3.18–3.44. However, with large input sizes it is likely that Equations 3.3–3.16 or Equations 3.18–3.44 may degrade performance significantly. To improve the execution time taken for large input it may be worth investigating the use of graphics processing units (GPU) to enhance Algorithm 3.2.2. If the GPU processing power can be used, large inputs sizes, such as the human genome, can be used as input to Opberg and results determined in a reasonable amount of time.

The approach of POS sequence alignment has some weaknesses. For one the algorithm does not take into account particles which may confer a different meaning than a sentence without the particle. An example of this would be the two sentences: *A does have a positive effect on B.* and *A does **not** have a positive effect on B.* Opberg does not take into account the negating word and a comparison would result in a high score. Another area for further research exploration is in entity extraction. Opberg may be good at finding similar structured sentences but in identifying the key terms, it is lacking. In real world applications, the authors envision to use Opberg in a pipeline with entity extraction methods run on the output after Opberg “filtering” for small labeled input sets.

Opberg represents a novel approach to causal discovery by considering alignments among POS mappings of sentences. This approach considers restrictions on the score size to break apart an alignment and enforces a minimum length requirement while also considering the number of alignments. OpBerg discovers meaningful alignments that return from alignment query results that are more useful in finding semantic similarity of two causal sentences. The improved model and efficient implementation make OpBerg the best model to use when performing tasks that involve the alignment of two or more sets of input, particularly in that of POS mappings for causal extraction.

3.3 ResearchMaps

Having extracted a causal representation, the next step is developing the semantics to describe the causal representations in a way that maximizes knowledge discovery. To establish this goal we seek to represent the extracted causality from Opberg into a causal network. We develop an application, called ResearchMaps which visualizes the network of biological assertions; a superset of information to causal assertions. Additionally, we formulate numeric descriptions of certainty that are in line with scoring heuristics of the neurobiologist. These numeric descriptions become the context for how we later describe casual assertions.

A research map is a directed graph that represents information concerning possible causal relations between biological phenomena [LS13]. Each node in the graph represents the identity and properties of a biological phenomenon, and each directed edge—from an *Agent* node to a *Target* node—represents a relation between phenomena (e.g., *A B*). In an experiment, an Agent is either intervened on or observed; this Agent may or may not act on another phenomenon, the Target, which is measured in the experiment. An Agent for one edge can be a Target for another. The key concepts captured in research maps reflect common epistemic practices in many fields of biology, represented in areas as diverse as neuroscience, development, immunology, and cancer. Thus, research maps should be useful to represent information in these and other areas of biology. See Figure 3.7 for an example of a research map of a published article [CAS16].

3.3.1 Framework

In research maps, biological experiments are categorized according to a hierarchical framework [SLB14]. We propose that experiments in many fields of biology can be classified into three general

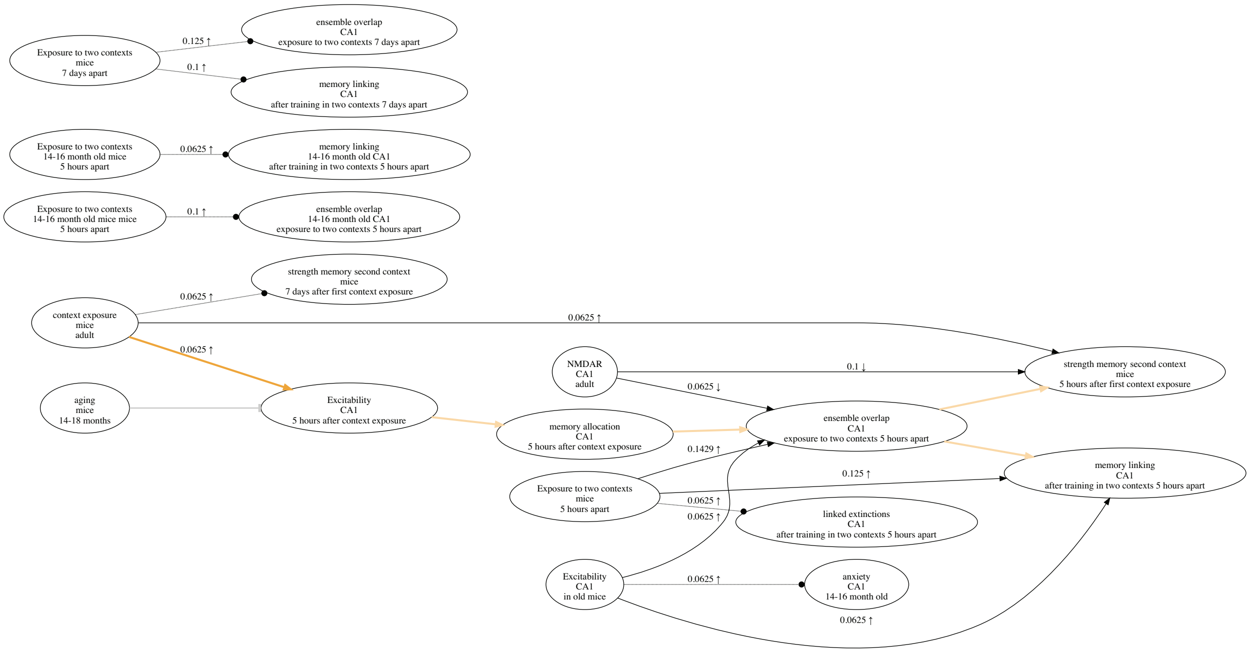


Figure 3.7: An example research map corresponding to the paper: "A shared neural ensemble links distinct contextual memories encoded close in time."

classes: (1) *Identity Experiments* attempt to identify phenomena and their properties; (2) *Connection Experiments*, the subject of research maps, test causal hypotheses; and (3) *Tool Development Experiments* develop and evaluate tools for performing Identity and Connection Experiments.

Within the class of Connection Experiments, we propose that there are four subclasses of experiments used to test a hypothesized connection between an Agent A and a Target B: (1) *Positive Intervention*, (2) *Negative Intervention*, (3) *Positive Non-intervention*, and (4) *Negative Non-intervention*. In a Positive Intervention experiment, the quantity or probability of the Agent A is increased, and the change (or lack of change) in Target B is measured. For example, to determine whether the activity of cell type A affects memory B, one could increase the activity of cell type A and then study the impact on memory B. In this case, the activity of cell type A is actively increased via an intervention—for instance, optogenetically.

A Negative Intervention experiment decreases the quantity or probability of A and measures B. For example, we could study how memory B is effected by a manipulation that inhibits cell type A. Positive and Negative Intervention experiments thus complement each other: the two use different approaches to probe the strength of the hypothesized connection between A and B.

Using only Positive and Negative Intervention experiments raises a number of problems that could confound the interpretation of those experiments. For example, such experiments always impose a change in an Agent A with methods that could have unintended effects. Therefore, any change observed in Target B may not necessarily result from a causal connection between A and B that is observable under specific conditions (e.g., during a spatial learning task); the change in B could instead be caused by experimental side effects of artificially intervening on A. The experimental process of intervening on A may inadvertently affect another phenomenon, C, even if C is not normally affected by A outside of the experimental setting. Although C may be the true cause of B, it may appear to the experimenter, who is oblivious to C's involvement, that A causes B. This possibility demonstrates the need for Non-intervention experiments to complement Positive and Negative Interventions.

A Non-intervention experiment measures A and B without intervening on either. In a Positive Non-intervention experiment, the quantity or probability of A is observed to increase, and the change (or lack of change) in B is measured. In a Negative Non-intervention experiment, the quantity or probability of A is observed to decrease, and B is measured. These experiments help us to learn whether the relation between A and B identified by Intervention experiments exists outside of the experimental setting used to intervene on A. Without Non-intervention experiments, it is difficult to be sure that experimental results are not mere artifacts caused by the interventions used to change A. In many fields of biology, Non-intervention experiments alone are usually judged to be insufficient to determine whether two phenomena are causally connected, as they are thought to merely document the correlation between these phenomena. However, elegant methods have been developed to identify specific causal structures solely from patterns of correlations derived from observational (i.e., non-interventional) data [SGS00].

From the four classes of experiments described above, we can glean evidence for three types of relations between phenomena. A relation between an Agent and a Target is defined as *Excitatory* when an increase in the Agent leads to an increase in the Target, or a decrease in the Agent leads to a decrease in the Target. In an Excitatory relation, a Positive Intervention experiment would result in an increase in the Target, and a Negative Intervention experiment would result in a decrease of the Target. In an *Inhibitory* relation, an increase in the Agent leads to a decrease in the Target, while a decrease in the Agent leads to an increase in the Target. When changes in the Agent fail to affect the Target, there is evidence for the absence of a connection between the two phenomena. In this last case, although the Agent and Target do not appear to be connected, this independence is represented explicitly with a relation denoted as *No-connection*.

3.3.2 Rules of Integration

In biology—and in research maps—a key approach to determine the reliability of results and the usefulness of hypotheses is to look for convergence and consistency in a set of findings. For instance, we can ask whether A reliably affects B or whether A and B are consistently independent of each other. We refer to the process that attempts to combine a series of experimental results as *Integration* [SLB14].

Integration methods determine the strength of the evidence for a particular connection, which is quantified and expressed as a score for a particular edge in a research map. The evidential strength of a connection is not to be confused with the magnitude of the causal effect that A has on B, where A may be one of many possible causes of B. Integration methods include (but are not limited to) *Convergence Analysis* and *Consistency Analysis* [SLB14]. By gauging the extent to which evidence is convergent and consistent, these Integration methods help to distinguish hypotheses with strong support from those with weak support. The principles of convergence and consistency are thus used for instantiating and scoring empirical edges in research maps.

Convergence Analysis assesses whether the outcomes of the different kinds of Connection Experiments (Positive and Negative Interventions, and Positive and Negative Non-interventions) are consistent with each other—i.e., whether they support a single connection type (either Excitatory, Inhibitory or No-connection). Suppose we find that optogenetically inhibiting cell type A is associated with a deficit in spatial learning. Suppose also that enhancing the activity of cell type A enhances the same form of learning. If we also found that cell type A is activated during spatial learning, and that this cell type is inactive when the animal is not learning, then our combined results would make a compelling argument that the activation of cell type A is causally connected to spatial learning. This convergence between these four classes of experiments would yield a relatively high score for the Excitatory connection in a research map representing the relation between cell type A

and spatial learning. On the other hand, contradictions among the data would lower the score of the connection. Convergence Analysis thus encompasses the notions that multiple lines of evidence are preferable to one, and that different experiment classes make unique contributions to testing the reliability of a hypothesized connection between two phenomena.

In addition to gauging the convergence of experimental results across multiple classes of experiments, it is also important to gauge the consistency of experimental results within each class of experiment. For this purpose, Consistency Analysis assesses whether experimental results are reproducible. For example, we might ask whether different kinds of Positive Interventions on the activity of cell type A (e.g., chemogenetic and optogenetic) always result in an enhancement of spatial learning. This question can refer to multiple iterations of the exact same experiment, or to a set of experiments that are similar in principle—e.g., two Positive Interventions of receptor A, one chemogenetic and the other optogenetic, that test two different forms of spatial learning.

3.3.3 Calculating Scores

To convey the amount of evidence for a particular empirical edge in a research map, a score for the edge is calculated using an algorithm based on the Integration methods above. These methods reflect epistemological rules and commonsense intuitions found in fields that use molecular and cellular approaches to biological problems, including neurobiology, biochemistry, cell biology, and physiology. In designing an approach to scoring such evidence, we strove to express quantitatively the following axioms: the principles of (1) convergence and (2) consistency, as described above; (3) the principle that convergence carries greater epistemological weight than consistency; and (4) the principle that we have no *a priori* reason to prefer one class of experiment to another when aggregating evidence. (In areas of science where one type of experiment is favored over others for technical reasons, our approach allows for a non-uniform weighting of evidence from different experiment classes.) There are other axioms used in science that have not been expressed in the scoring algorithm of research maps [SLB14] because we see them to be secondary to the ones above.

The central idea of our scoring approach is that convergent and consistent results increase the score of an edge, while conflicting results decrease the score. Each score falls in the range $(0,1)$, and each experiment class (Positive Intervention, Negative Intervention, Positive Non-intervention, and Negative Non-intervention) contributes an amount in the range $(0,0.25)$ to the overall score. Multiple experiments of the same kind contribute progressively smaller scores to the edge. As experiments are recorded, a Bayesian approach is used to update the degrees of belief attributed to each type of relation. The scores thus reflect an approach for gauging the strength of the convergent and consistent evidence supporting a given connection; their semantics are derived not from their absolute values but from their relative values. In addition to p-values from statistical tests and associated meta-analyses, this scoring method could conceivably be used to evaluate the strength of evidence across various types of experiments testing a single causal assertion.

The score for an edge in a research map is calculated as follows. Let $C = \left\{\uparrow, \varnothing^{\uparrow}, \varnothing^{\downarrow}, \downarrow\right\}$ denote the set of all experiment classes, where $c = \uparrow$ denotes the class Positive Intervention; $c = \varnothing^{\uparrow}$ denotes the class Positive Non-intervention; $c = \varnothing^{\downarrow}$ denotes the class Negative Non-intervention; and $c = \downarrow$ denotes the class Negative Intervention. Let $R = \{\mathcal{E},\mathcal{N},\mathcal{I}\}$ denote the set of relations that can exist between two phenomena and for which an experiment can provide evidence, where $\mathcal{E}$ denotes an Excitatory relation; $\mathcal{N}$ denotes a No-connection relation; and $\mathcal{I}$ denotes an Inhibitory relation. Thus, an experiment of class $c \in \left\{\uparrow, \varnothing^{\uparrow}, \varnothing^{\downarrow}, \downarrow\right\}$ can yield evidence in support of relation $r \in \{\mathcal{E},\mathcal{N},\mathcal{I}\}$.

$$\overrightarrow{\alpha}_c = (\alpha_{c,\mathcal{E}},\alpha_{c,\mathcal{N}},\alpha_{c,\mathcal{I}}) \tag{3.74}$$

$$\overrightarrow{\theta}_c = (\theta_{c,\mathcal{E}},\theta_{c,\mathcal{N}},\theta_{c,\mathcal{I}}) \tag{3.75}$$

$$\overrightarrow{x}_c = (x_{c,\mathcal{E}},x_{c,\mathcal{N}},x_{c,\mathcal{I}}) \tag{3.76}$$

$$(\theta_{c,\mathcal{E}},\theta_{c,\mathcal{N}},\theta_{c,\mathcal{I}}) \sim \text{Dir}(\alpha_{c,\mathcal{E}},\alpha_{c,\mathcal{N}},\alpha_{c,\mathcal{I}})\,, \tag{3.77}$$

$$(x_{c,\mathcal{E}},x_{c,\mathcal{N}},x_{c,\mathcal{I}}) \sim \text{Mult}(\theta_{c,\mathcal{E}},\theta_{c,\mathcal{N}},\theta_{c,\mathcal{I}},n_c)\,. \tag{3.78}$$

Here, $\alpha_{c,r}$ is the prior weight given to relation r supported by experiments of class c; $\theta_{c,r}$ is the probability that the next experiment of class c will yield evidence in support of relation r; $x_{c,r}$ is the number of experiments of class c that have yielded evidence in support of relation r, and n_c is the number of experiments of class c that have been performed. For each class of experiment c, we can

	$B+$	$B0$	$B-$
$A\uparrow$	5	1	1
$A\varnothing^{\uparrow}$	1	1	2
$A\varnothing^{\downarrow}$	1	1	2
$A\downarrow$	1	1	1

$$\mathrm{P}(\mathcal{E}) = (1/4)\left[(5/7)_{\uparrow} + (1/4)_{\varnothing\uparrow} + (2/4)_{\varnothing\downarrow} + (1/3)_{\downarrow}\right] = 0.449$$

$$\mathrm{P}(\mathcal{N}) = (1/4)\left[(1/7)_{\uparrow} + (1/4)_{\varnothing\uparrow} + (1/4)_{\varnothing\downarrow} + (1/3)_{\downarrow}\right] = 0.244$$

$$\mathrm{P}(\mathcal{I}) = (1/4)\left[(1/7)_{\uparrow} + (2/4)_{\varnothing\uparrow} + (1/4)_{\varnothing\downarrow} + (1/3)_{\downarrow}\right] = 0.307$$

Relation = Excitatory **Score** = $\dfrac{0.449 - (1/3)}{1 - (1/3)} = 0.174$

Figure 3.8: An example of a score calculation given the observed values in the table.

define $\overrightarrow{x}_c$ (compare to the table in Figure 3.8):

$$\overrightarrow{x}_{\uparrow} = [x_{\uparrow,\mathcal{E}}, x_{\uparrow,\mathcal{N}}, x_{\uparrow,\mathcal{I}}] \tag{3.79}$$

$$\overrightarrow{x}_{\varnothing\uparrow} = \left[x_{\varnothing\uparrow,\mathcal{E}}, x_{\varnothing\uparrow,\mathcal{N}}, x_{\varnothing\uparrow,\mathcal{I}}\right] \tag{3.80}$$

$$\overrightarrow{x}_{\varnothing\downarrow} = \left[x_{\varnothing\downarrow,\mathcal{E}}, x_{\varnothing\downarrow,\mathcal{N}}, x_{\varnothing\downarrow,\mathcal{I}}\right] \tag{3.81}$$

$$\overrightarrow{x}_{\downarrow} = [x_{\downarrow,\mathcal{E}}, x_{\downarrow,\mathcal{N}}, x_{\downarrow,\mathcal{I}}] \tag{3.82}$$

The score of an edge is based on the values of θ_c for each of the experiment classes, which are updated as additional experiments are recorded, thereby changing the values of $\overrightarrow{x}_c$. We are thus interested in estimating each $\overrightarrow{\theta}_c$ in light of the evidence represented by each $\overrightarrow{x}_c$. Applying Bayes theorem yields

$$p(\overrightarrow{\theta}_c \mid \overrightarrow{x}_c, \overrightarrow{\alpha}_c) \propto \theta_{c,\mathcal{E}}^{\alpha_{c,\mathcal{E}}+x_{c,\mathcal{E}}-1} \theta_{c,\mathcal{N}}^{\alpha_{c,\mathcal{N}}+x_{c,\mathcal{N}}-1} \theta_{c,\mathcal{I}}^{\alpha_{c,\mathcal{I}}+x_{c,\mathcal{I}}-1} \tag{3.83}$$

The posterior distribution is in the form of a Dirichlet distribution, so we have that

$$\theta_c \mid \overrightarrow{x}_c, \alpha_c \sim Dir(\alpha_c + \overrightarrow{x}_c)\,. \tag{3.84}$$

The expected value of this distribution is thus expressed as

$$E\left[\theta_{c,r} \mid \overrightarrow{x}_c, \alpha_c\right] = \frac{\alpha_{c,r} + x_{c,r}}{\sum_r \alpha_{c,r} + n_c}\,. \tag{3.85}$$

If $\alpha_{c,r} = 1$ for all c and r, the above expression becomes

$$E\left[\theta_{c,r} \mid \overrightarrow{x}_c, \alpha_{c,r} = 1\right] = \frac{1 + x_{c,r}}{|R| + n_c}, \tag{3.86}$$

which is an implementation of Laplace (add-one) smoothing.

In the absence of evidence (i.e., before any experiments are performed), $x_{c,r} = 0$ for all c, r. We denote this state by θ_o:

$$\theta_o = E\left[\theta_{c,r} \mid \overrightarrow{x}_c = (0,0,0), \alpha_{c,r} = 1\right] = \frac{1}{|R|} = \frac{1}{3}. \tag{3.87}$$

Let $\overline{\theta}$ denote the set of mean r-components across all experiment classes (an expression of convergence):

$$\overline{\theta}_{\mathcal{E}} = \sum_c E\left[\theta_{c,\mathcal{E}} \mid \overrightarrow{x}_c, \alpha_{c,\mathcal{E}} = 1\right] \tag{3.88}$$

$$\overline{\theta}_{\mathcal{N}} = \sum_c E\left[\theta_{c,\mathcal{N}} \mid \overrightarrow{x}_c, \alpha_{c,\mathcal{N}} = 1\right] \tag{3.89}$$

$$\overline{\theta}_{\mathcal{I}} = \sum_c E\left[\theta_{c,\mathcal{I}} \mid \overrightarrow{x}_c, \alpha_{c,\mathcal{I}} = 1\right] \tag{3.90}$$

$$\overline{\theta} = \frac{1}{|C|}\left[\overline{\theta}_{\mathcal{E}}, \overline{\theta}_{\mathcal{N}}, \overline{\theta}_{\mathcal{I}}\right] \tag{3.91}$$

The relation assigned to the research-map edge is the relation with the largest component in $\overline{\theta}$:

$$\underset{r}{\operatorname{argmax}}\, \overline{\theta}_r \tag{3.92}$$

The score assigned to the research-map edge is

$$\frac{\max \overline{\theta} - \theta_o}{1 - \theta_o} \tag{3.93}$$

where $\max \overline{\theta}$ denotes the largest component of $\overline{\theta}$. In cases where two or more components of $\overline{\theta}$ are equal, neither a relation nor a score is assigned to the edge.

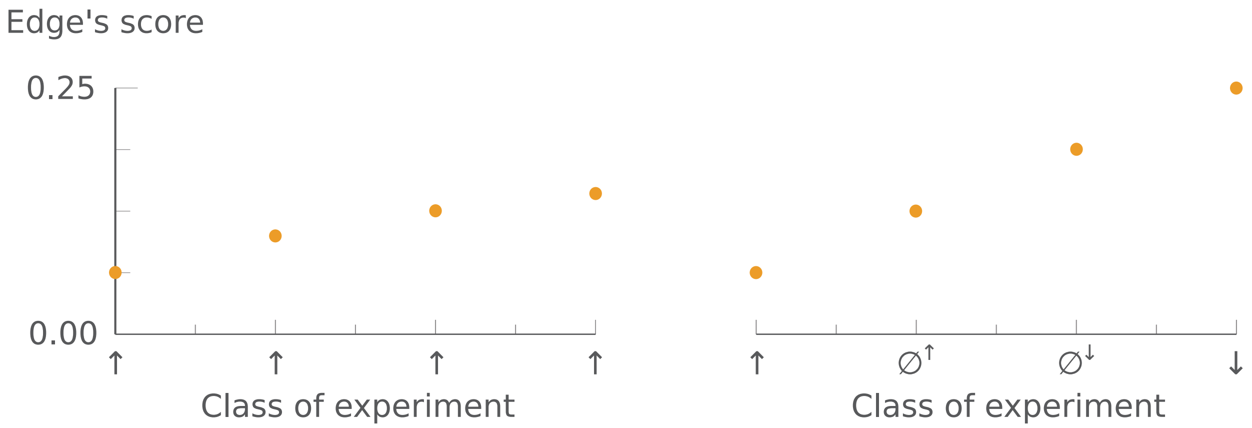


Figure 3.9: Plots showing how the edge score value changes with increasing the number of experiment of the same class (left) and with different experiment classes (right).

See Figure 3.8 for a depiction of a shorthand calculation of an edge's score. See Figure 3.9 for plots of how the score of an edge increases with each subsequent experiment due to the principles of consistency and convergence.

It is worth noting that the scores derived from the above scoring algorithm, which is based on Bayesian principles, closely resemble those derived from another heuristic scoring approach from early versions of research maps, which expressed scientists' intuitions regarding the integration of evidence [SM15].

3.3.3.1 A scoring example

To develop an intuition for the above scoring approach, consider the following example, which uses the experiments involving CREB and the number of Arc neurons that are depicted in Figure 3.10(a). In this research map, the edge connecting these two nodes represents three experiments: two Positive Interventions of CREB resulting in no change in the number of Arc neurons, and one Negative Intervention of CREB, again resulting in no change. Together, these three experiments provide evidence for a No-connection edge between the two nodes. Before any of these experiments were performed, $\overrightarrow{\theta_c}$ was uniform for all c. After the first experiment, in which a Positive Intervention produced no change in the Target, $\overrightarrow{\theta_c} = (0.25{,}0.50{,}0.25)$ and the score of the edge was 0.0625.

After the second Positive Intervention (with the same result as the first), the score of the edge became 0.1000.

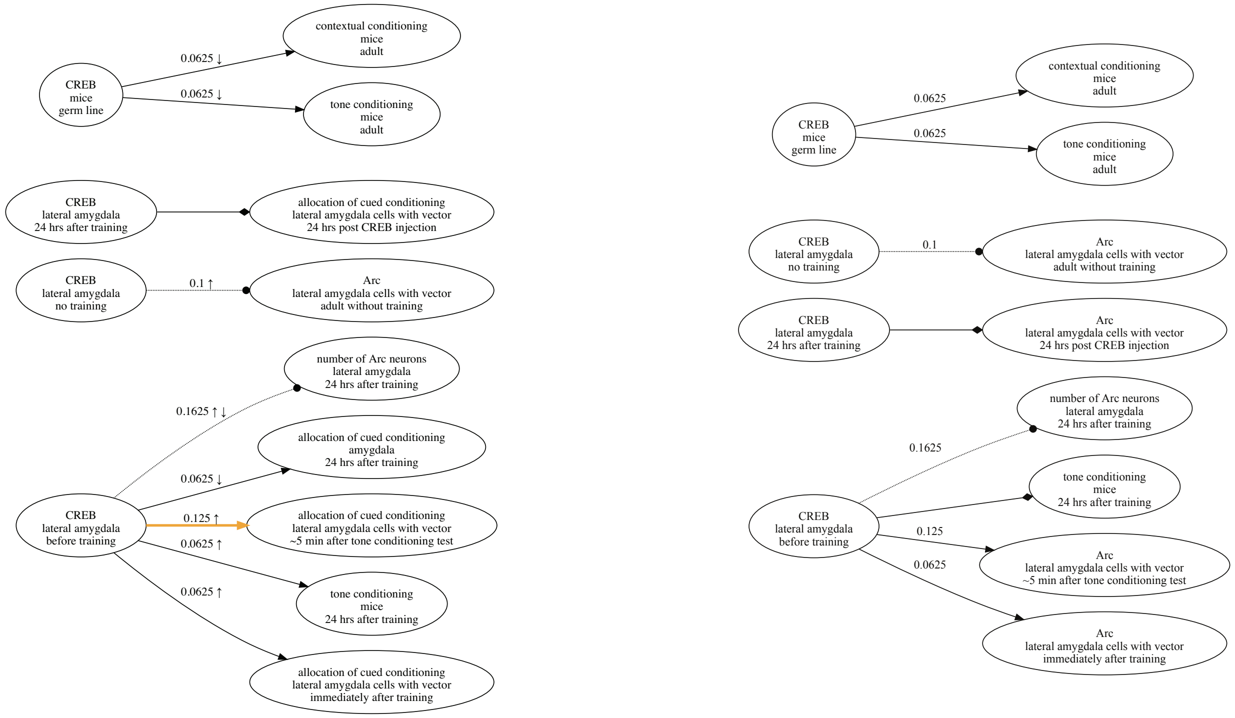
The first Positive Intervention thus changed the score by 0.0625, while the second experiment changed the score by 0.0375. These two changes in the score demonstrate a commonsense intuition regarding evidence that is expressed quantitatively by the scoring algorithm: each subsequent experiment that yields consistent results increases the score, albeit by an amount that is less than the amount contributed by the previous consistent experiment.

After the third experiment, in which a previously unrepresented experiment class (Negative Intervention) yielded a consistent result (no change), the score increased to 0.1625, for a net change of 0.0625. This change demonstrates another desirable feature of the scoring algorithm: when consistent results are obtained across multiple experiment classes, each sequence of experiments within a class contributes the same set of decaying amounts to the score, such that results across the four experiment classes are weighted independently of the order in which they were obtained.

If a fourth experiment with conflicting evidence were recorded—for example, a Positive Non-intervention yielding an increase in the Target—the score would drop to 0.1313. Appropriately, the conflicting evidence would undermine the still-dominant evidence that the relation between the two nodes is No-connection. Had this conflicting evidence come from another Positive Intervention, an experiment class already represented in the score, the score would drop to 0.1250. This larger drop than the one incurred for a conflicting Positive Non-intervention reflects the idea that scientists tend to trust evidence from a particular experiment class to the extent that experiments within this class yield consistent results.

3.3.4 Components of ResearchMaps

In ResearchMaps, an Agent or Target is defined in three complementary ways: *what* the phenomenon is, *where* the phenomenon exists, and *when* the phenomenon acts. ResearchMaps stores this information as three properties for each node: (1) `What` describes a key identifier of the phenomenon involved (e.g., the name by which the gene, protein, cell, organ, behavior, etc. is known); (2) `Where` describes the location of the `What` (e.g., the organ, species, etc.); and (3) `When` provides temporal information that is critical to the identity of the `What` (e.g., the time, age, phase, etc.). For example,


CREB mice germ line
0.0625 ↓
contextual conditioning mice adult
0.0625 ↓
tone conditioning mice adult
CREB lateral amygdala 24 hrs after training
allocation of cued conditioning lateral amygdala cells with vector 24 hrs post CREB injection
CREB lateral amygdala no training
0.1 ↑
Arc lateral amygdala cells with vector adult without training
number of Arc neurons lateral amygdala 24 hrs after training
0.1625 ↑ ↓
allocation of cued conditioning amygdala 24 hrs after training
0.0625 ↓
CREB lateral amygdala before training
0.125 ↑
allocation of cued conditioning lateral amygdala cells with vector ~5 min after tone conditioning test
0.0625 ↑
tone conditioning mice 24 hrs after training
0.0625 ↑
allocation of cued conditioning lateral amygdala cells with vector immediately after training
CREB mice germ line
0.0625
contextual conditioning mice adult
0.0625
tone conditioning mice adult
CREB lateral amygdala no training
0.1
Arc lateral amygdala cells with vector adult without training
CREB lateral amygdala 24 hrs after training
Arc lateral amygdala cells with vector 24 hrs post CREB injection
number of Arc neurons lateral amygdala 24 hrs after training
0.1625
CREB lateral amygdala before training
tone conditioning mice 24 hrs after training
0.125
Arc lateral amygdala cells with vector ~5 min after tone conditioning test
0.0625
Arc lateral amygdala cells with vector immediately after training

(a) (b)

Figure 3.10: A research map of experiments involving the transcription factor CREB (a) alongside the initial map of experiments exploring the role of CREB in amygdala memory enhancements (b).

if the protein neurofibromin is measured in multiple locations, a corresponding research map would include multiple nodes for neurofibromin with different `Where` properties. This approach is instructive, as neurofibromin could have different biological characteristics in different cellular locations (e.g., excitatory neurons versus inhibitory neurons) or at different stages of development. ResearchMaps displays the `What`, `Where`, and `When` properties on separate lines within each node.

In ResearchMaps, the four experiment classes are represented by symbols above each empirical edge. As given in set C above, Positive Interventions are represented by an upward arrow ($\uparrow$); Negative Interventions are represented by a downward arrow ($\downarrow$); Positive Non-interventions are represented by the empty set symbol and a superscript upward arrow ($\varnothing^{\uparrow}$); and Negative Non-interventions are represented by the empty set symbol and a superscript downward arrow ($\varnothing^{\downarrow}$). Although we have not yet defined a formal representation for experiments involving more than two nodes, ResearchMaps accommodates intervention experiments with two Agents. At the time of this writing, such experiments comprise approximately fourteen percent of the experiments logged. The putative mechanisms underlying the results of these multi-intervention experiments can be visualized using hypothetical edges among the three entities involved (two Agents and one Target); the structure of these hypothetical edges is provided by the user.

ResearchMaps can accommodate information about the statistical test used to establish each finding and its associated p-value. Such information is of course valuable in evaluating experiments; however, as the areas covered by research maps are diverse, and there are no standards as to which statistics are used and how to report them, p-values do not currently affect the score of research-map edges, and they are optionally tracked by each user. See Figure 3.7 for an example of a research map.

3.3.4.1 Empirical and hypothetical edges

ResearchMaps allows the user to input both empirical and hypothetical edges between any two phenomena (and, by extension, empirical and hypothetical nodes). A hypothetical edge represents a putative connection with no direct experimental evidence. Hypothetical edges are usually implied by empirical edges, and they are often key in interpreting and reporting the results of a research article. Since hypothetical edges do not represent empirical evidence, they are assigned neither scores nor experiment symbols. To visually differentiate hypothetical edges, they are shown in a lighter color and without a score or experiment symbols.

Beyond allowing users to track various hypotheses, hypothetical edges can also help to structure research maps of empirical evidence, as illustrated with the following example. Consider a signaling pathway (e.g., a biochemical cascade), which we will represent as $A{\rightarrow}B{\rightarrow}C{\rightarrow}D$. Just as hypotheses help to frame and organize the results of research articles, hypothetical edges help to structure and contextualize empirical edges in a research map. For instance, a map that represents the connections $A{\rightarrow}C$, $A{\rightarrow}D$, and $B{\rightarrow}D$ (Figure 3.11) would not explicitly reflect the putative $A{\rightarrow}B{\rightarrow}C{\rightarrow}D$ pathway because not all connections in this pathway are part of that map. By including in the resulting map the hypothetical edges $A{\rightarrow}B$, $B{\rightarrow}C$, and $C{\rightarrow}D$, the underlying hypothesis for the experiments carried out is immediately obvious (Figure 3.11).

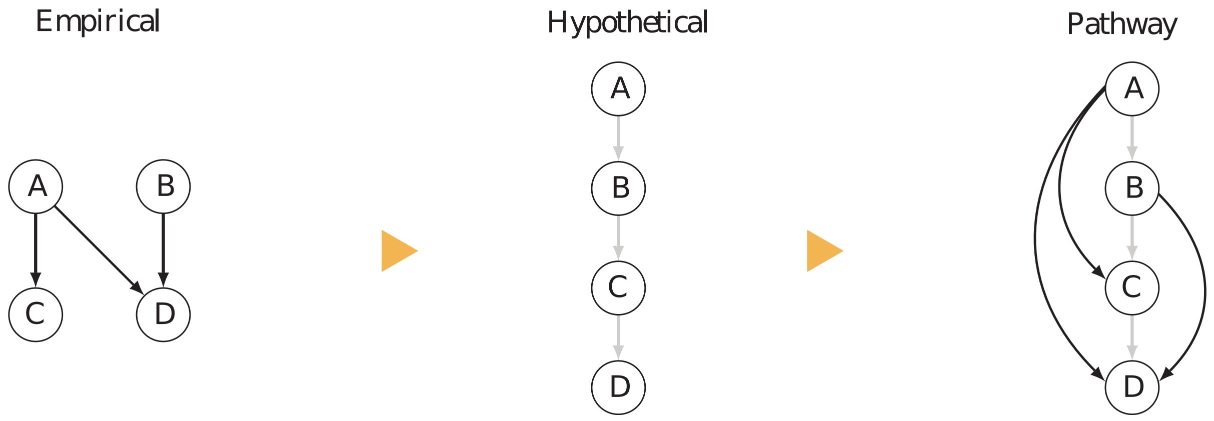


Figure 3.11: A diagram showing how hypothetical edges (in gray) help to organize empirical edges in a research map, thus framing the empirical results with a specific hypothesis.

3.3.5 Generating maps for research articles

There are multiple steps to make a research map for a given research article. The first step is to identify all the nodes that will be included in the research map. This process entails the identification of Agent–Target pairs involved in the reported experiments. For any one Agent–Target pair, the next step is to find the experiment class that was performed to test their relation. In addition to the class of the experiment, the user can record the result that was obtained, as well as the key techniques that were used to observe (or manipulate) the Agent and observe the result in the Target. Once the empirical edges are entered (ones for which an experiment is reported), any hypothetical edges suggested by the article can be added, thereby helping to structure the map and contextualize the empirical results. Finally, because research maps can become large and complex, it is instructive to highlight the main connections, whether they are hypothetical or empirical.

3.3.6 Research maps at work

As stated above, research maps are designed to facilitate the personal curation of information derived from detailed analyses of research articles, reviews, and other sources central to the activities of scientists. The derived maps are designed to function at the interface between the large body of information that could potentially be relevant to any one individual scientist, and that subset of empirical and hypothetical assertions that an individual scientist judges to be directly relevant to ongoing work. For example, in the space of three years, one of our users created public research maps for 125 articles with 2,251 experiments, 1,293 nodes, and 1,693 edges. Even in this relatively small set of articles, the sheer number of empirical and hypothetical relations is too large for most individual scientists to remember, objectively integrate, and systematically reason through.

Additionally, the process of mapping information critical for a project affords a clarity that is harder to come by any other way. For instance, a few years ago some of us were involved in experiments that suggested that the expression of the cAMP responsive element binding (CREB) transcriptional factor in a small subset of neurons in the lateral amygdala of mice could lead to enhancements of memory for both auditory and contextual fear conditioning. These results were surprising, and they led to a series of experiments that explored the nature of these memory enhancements. One of the motivations for these experiments was the hypothesis that the cellular levels of CREB may be one factor that determines the subset of lateral amygdala neurons that go on to store a given fear memory [HKY07]. The initial research map of the experiments designed to explore the CREB memory enhancement is shown in Figure 3.10(b). While thinking of the connections in that article with the help of research maps, we realized that there may be a more fundamental concept that could both provide a better structure for the map and a more useful framework for future experiments (Figure 3.10(a)).

In our initial experiments [HKY07], we used positive and negative manipulations of CREB, and determined which lateral amygdala cells were involved in memory by using the immediate early gene Arc, a gene whose expression is thought to tag cells involved in memory [GMB99]. Mapping these findings helped us to realize that we needed to identify a phenomenon that captured the idea that CREB was instrumental in determining which cells were involved in memory. To this end, we borrowed a term from computer science—*memory allocation*—and the process of defining this new neuroscience phenomenon in our research maps also helped us to identify the need for other methods to measure it [ZWK09].

Research maps also helped us to focus our attention on the mechanisms by which CREB modulated memory allocation [HKY09, ZWK09] and aided us in defining a research plan to tackle this new complex problem. Although it is possible that we and others could have arrived at similar research decisions without the help of research maps, the ability to precisely map information imparted a degree of clarity that helped us to think through these experiments and develop our

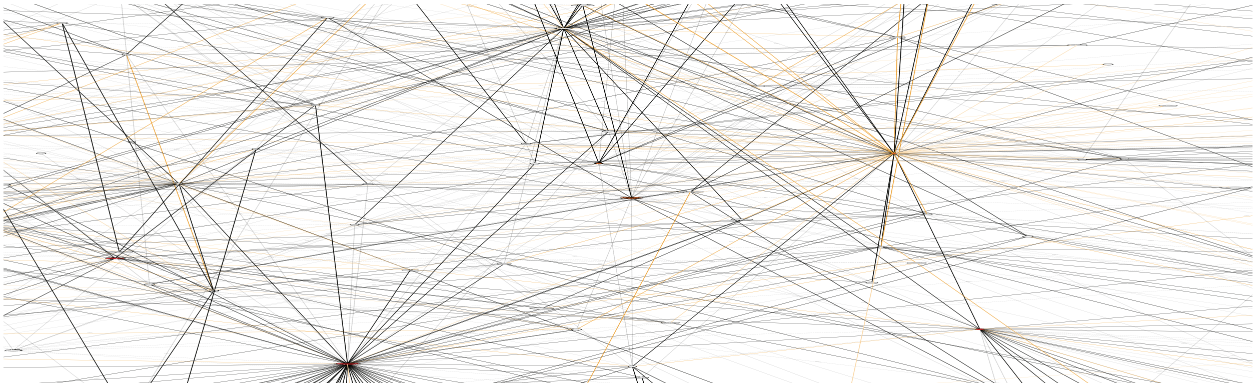


Figure 3.12: Global research map of experiments in memory allocation and other related work.

past and current research on memory allocation. The concept of memory allocation [FJ15] that emerged out of these efforts led to a number of research articles [HKY09, ZWK09, SMY13, KKK14, YMY14] that explored the mechanistic basis of this concept and tested its possible role in other brain structures, such as the insular cortex [SSZ14] and in processes such as memory linking [CAS16, RYM16].

When reading new research articles, the underlying mechanisms are not always apparent. However, in our experience, the process of extracting and formalizing information about possible connections tested in these articles has always enhanced our understanding of the reported findings. This formalization process also brings the information from disparate articles into a shared framework that facilitates integration of this information, as well as experiment planning.

Using research maps to visualize our work in memory allocation has also provided insight into how these experiments are connected to research in related areas. Figure 3.12 shows a research map of our work in memory allocation and all other research maps of articles that connect to it. Figure 3.13 shows a bar graph indicating the number of nodes in our ResearchMaps database that are connected to nodes pertaining to work in memory allocation. Analysis of the data represented in these two figures suggests that research maps provide a rich platform in which to generate and evaluate hypotheses about the mechanisms that may be modulating memory allocation.

3.4 Experiment Planning

With the semantics around representing a superset of causal information established, we seek to make meaningful discoveries from the biological assertions synthesized into the causal form. One area we explore is that of experiment planning. Given the prior network of biological relationships synthesized into causal expressions, we investigate the ability to plan experiments through causal mechanisms and discovery.

A major goal in science is to identify causal mechanisms. Scientists try to understand, for instance, how cigarettes cause lung cancer, or how a genetic mutation causes memory loss. As suggested by the refrain "correlation does not equal causation," a causal model not only predicts correlations in a system but also predicts how that system will respond under interventions. This difference between correlative and causal models is particularly crucial for a physician, who tries to cure a patient's disease with a surgical or pharmacological intervention.

In the last few decades, causality has been formalized using mathematics, yielding the enormously successful model known as a causal Bayesian network, or causal Bayes net [SGS00, Pea09]. This model represents causality with a *causal graph*, a network of nodes and directed edges (e.g., $X \rightarrow Y$) that correspond to the system's variables and causal relations. Using this model of causality, researchers have developed *causal discovery* algorithms, which identify the causal graph that describes and predicts the behavior of a system's variables [Ebe17].

There are a variety of causal discovery algorithms that operate on primary data [MD18]; however, there has been relatively little work on the problem of building causal models with only textual information from scientific communication. This is an important problem because much of the information that a scientist encounters is free text: research articles, for instance, are often unaccompanied by primary data but contain aggregate statistics that should inform a scientist's understanding of the system.

To address this problem, we present a pipeline for *meta-analytic causal discovery*: First, the scientist annotates statistical results in free-text research articles—for instance, using the representation given in Section 3.3. Next, these annotations are input to an algorithm that identifies the causal graphs consistent with the annotated results. The scientist can then inspect the consistent

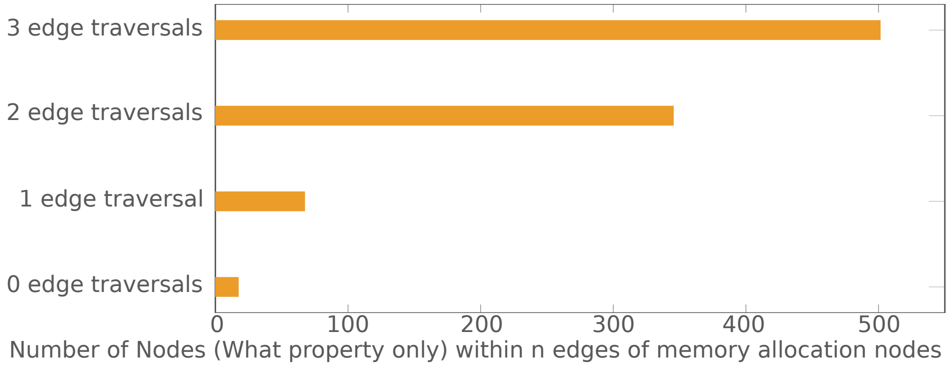


Figure 3.13: Connectivity characteristics in the global map for memory allocation.

graphs to see which inferences arise out of the synthesis of annotated research articles [MWW17a, MWW17b]. In our proposed technique, we demonstrate how this meta-analytic approach can inform not only evidence synthesis but also experiment selection.

In biology, selecting the next experiment often requires causal reasoning. Biologists must examine the evidence and identify logically consistent explanations, which may agree in some respects but disagree in others, depending on the amount of evidence. Based on this analysis, biologists hypothesize a causal mechanism and select an experiment to test it. With the primary data from studies, biologists can use causal discovery algorithms [Ebe17] to identify causal mechanisms. These methods have even motivated formal approaches to experiment selection [Mur01, EGS05, MML05, Ebe08, HG08, HB12, HHE13]. But biologists often do not have access to primary data; instead, they rely on literature, rendering many of these causal discovery methods unusable.

We seek to generalize a causal discovery method to a meta-analytic technique that can integrate multiple forms of causal information, including qualitative knowledge from literature. We allow our method to input annotated empirical results from research articles to automatically derive *every* consistent causal interpretation, expressed as a set of causal graphs [SGS00, Pea09]. These graphs synthesize the causal implications of empirical results and provide a formal, hypothesis-generating device for selecting experiments: they show precisely which relations are determined, and which remain underdetermined. We can then use a "degrees-of-freedom" analysis that concisely visualizes features of these consistent explanations.

Causal graphs are similar to biological pathway diagrams, but they have mathematical properties that make them more suitable for synthesizing empirical results. Like a pathway diagram, a causal graph shows a system's phenomena and relations between them. But a causal graph also encodes specifically *causal* relations and some of their statistical properties; it can thus visualize a system intuitively while still encoding precise and predictive mathematical relationships—a feature that is absent in many pathway diagrams. Consider, for example, Figure 3.14, which shows a pathway diagram from the biological literature. Although it clearly illustrates biological mechanisms, its edges lack precise mathematical definitions; different edges can have different semantics, and the overall diagram thus does not provide one clear interpretation. This ambiguity is compounded if a researcher tries to synthesize multiple pathway diagrams.

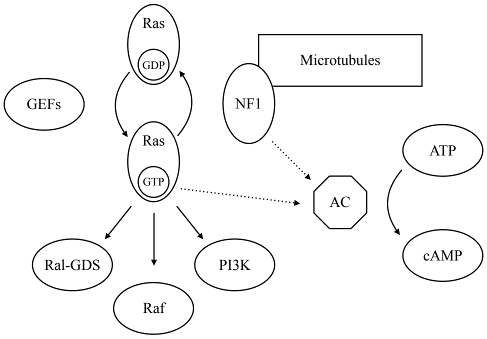


Figure 3.14: An example of a pathway diagram that has been adapted from the literature [CS03]. This diagram illustrates biological mechanisms, but because the meaning of each edge is not precisely defined, this diagram cannot necessarily be used to reason causally about the system.

If qualitative pathway diagrams from different articles are simply "stitched" together—by overlapping common nodes and pooling all the diagrams' edges—the hybrid diagram may bias researchers, inviting them to reify specific pathways that the evidence does not support, or that the evidence even contradicts. For instance, Figures 3.15(a) and 3.15(b) are typical of pathway diagrams in the biological literature; they are not formal causal graphs but rather illustrations in which $X \rightarrow Y$ implies that a change in X was observed to precede a change in Y, ostensibly implying a causal interaction. Note the consequence of stitching these two diagrams together (Figure 3.15(c)): through the $X \rightarrow Z$ edge, it appears as if X can affect Z independently of Y. But that is not necessarily true. It's possible, for instance, that in the experiment that led to Figure 3.15(a), Y was simply unmeasured; in this case, Y still could have mediated X's effect on Z, but this mediation may have been unknown to the researchers who instead focused on X and

Z. This sort of bookkeeping can become quite complicated, even for a small system. And these diagrams' imprecise semantics impede the development of an algorithmic solution to this problem.

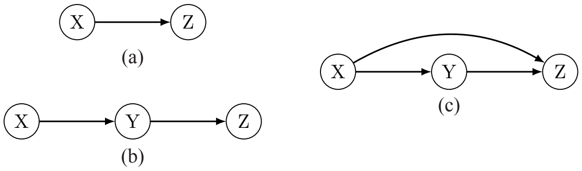


Figure 3.15: Pathway diagrams from the literature cannot simply be "stitched" to derive causal inferences of empirical results. When the nodes and edges from (a) and (b) are simply pooled to produce (c), this new diagram suggests that, via the $X \to Z$ edge, X can effect Z independently of Y—an interpretation that does not necessarily follow from the empirical evidence that led to (a) and (b).

In contrast, the theory of causality gives a principled procedure for stitching causal graphs, ensuring that the hybrid model is consistent with the evidence. This stitching process resembles what scientists do when they try to synthesize the empirical evidence in research articles. Considering the enormity of the space of causal graphs, biologists would benefit from software that automatically computes the causal implications of a set of findings.

In much of the causal discovery literature, it is assumed that an experiment allows scientists to observe every variable in the system simultaneously. However, this is often infeasible: instead, scientists perform experiments on subsets of the system's variables and combine the results from these subsets analytically—a technique known as *piecemeal causal discovery* [May11, May14, May19]. This approach is often required in fields like biology due to technological limitations and living organisms' immense complexity. Piecemeal causal discovery often fails to identify the one true causal graph for the system under investigation, regardless of the number of experiments that can be performed [May11, May14, May19].

In the context of piecemeal causal discovery, we conceive of experiment selection as encompassing two main decisions: (1) the choice of which phenomena—out of all potential phenomena in a system—will be involved in the experiment, and (2) the choice of which empirical strategy will be used: either a passive observation or an intervention where one or more of the phenomena are manipulated. Here we consider studies that each involve two phenomena, where neither or one of the phenomena is intervened on—a widespread occurrence in molecular and cellular biology as detailed in Section 3.3. For instance, given the available evidence and a limitation on the number of variables that can be observed simultaneously, it may be more informative to intervene on variable X and observe the response of variable Y than it would be to intervene on X and observe Z; in other situations—with different evidence available—the opposite may be true. Consider, for example, the following three causal graphs:

1. $X \rightarrow Y, X \rightarrow Z$

2. $X \leftarrow Y, X \rightarrow Z$

3. $X \rightarrow Y, X \leftarrow Z$

If we obtained information that led us to believe the true causal graph was either graph 1 or graph 2, it would be more informative to intervene on X and observe whether Y covaried, thus allowing us to determine the relation between X and Y. (Note that graphs 1 and 2 have the same edge relation between X and Z.) If instead, we obtained information that led us to consider graph 1 and graph 3, we would then prefer the experiment in which we intervene on X and observe whether Z covaries, as these two graphs have the same edge relation between X and Y. There are still other situations where, in the presence of conflicting evidence, it could be most instructive to repeat an experiment. These decisions are often left to the subjective judgement of the scientist [Ebe10]. A more objective and systematic approach is achieved by representing empirical results with causal graphs.

Causal discovery algorithms will often return not a single causal graph but a set of graphs, each of which equally satisfies the constraints imposed by the input data. This set of consistent causal graphs is known as a *(Markov) equivalence class* [SGS00]. The size of the equivalence class indicates the number of causal explanations that remain viable, given what is known; it thus indicates our degree of ignorance regarding the system. Therefore, an equivalence class not only synthesizes the causal implications of empirical evidence but also provides a formal, hypothesis-generating device for selecting experiments: it encodes precisely which causal relations are determined, and which relations remain underdetermined. For instance, if a set of empirical results is consistent with more than one causal graph—each with its own configuration of edges—a researcher can assess which hypotheses are worth pursuing by inspecting exactly which causal relations remain viable. A causal graph's underdetermination can thus help researchers to plan experiments by indicating which experiments are needed to fully determine the causal structure of the system.

We characterize this underdetermination with a causal graph's *degrees of freedom*, which represent the diversity of edge relations that appear in the graphs of an equivalence class [MWW17a, MWW17b]. For example, all graphs in an equivalence class may have the same edge relation between the variables X and Y (e.g., $X \rightarrow Y$), but there may be a diversity of edge relations between the variables Y and Z (e.g., $Y \leftarrow Z$ and $Y \rightarrow Z$). In light of these options, potential experiments can be chosen based on how much information they would provide—specifically, how much they could distinguish between remaining relations, thus pruning the existing model space of consistent causal graphs. This analysis must be agnostic to the result of each potential experiment, which of course cannot be known in advance. With simulations, we show that experiment selection based on the equivalence class's degrees of freedom outperforms random experiment selection, in that fewer experiments are needed to identify causal structures. Within the same computational framework, we also demonstrate how to categorize a given hypothesis according to its utility for revealing new causal information regarding the system under investigation.

This approach thus makes the following contributions:

1. Two experiment-selection algorithms with readily interpretable heuristics tailored to meta-analytic piecemeal causal discovery—a setting that is ubiquitous in the biological sciences (Sections 3.4.1.1–3.4.1.2);

2. Simulations of the experiment-selection algorithms that demonstrate (1) trade-offs between computational efficiency and the efficiency of experimentation for causal discovery, as well as (2) inherent limitations of piecemeal causal discovery involving two-variable experiments (Section 3.4.2);

3. A hypothesis-categorization algorithm that guarantees whether an experiment designed to test a given hypothesis could possibly yield new causal information that would further determine a system's causal structure, given a knowledge base of existing experimental results (Section 3.4.1.3);

4. A simulation of the hypothesis-categorization algorithm that demonstrates how the proportion of informative and uninformative hypotheses changes as causal-structure information is obtained through experimentation (Section 3.4.2).

3.4.1 Approach

Given a set of (in)dependence relations expressed as constraints on causal structure, we use the causal discovery algorithm discussed above to obtain the degrees of freedom for the equivalence class of causal graphs that are consistent with the constraints. For the case where we assume that the true causal graph is a DAG, the approach is given by Algorithm 3.4.2 and proceeds as follows. We define the set $\mathbf{K}$ as the set of causal-structure constraints obtained for a system with the set of variables $\mathbf{V}$. For each $\{X,Y\} \in \mathbf{V}$, we query the SAT solver once for every degree of freedom that can exist between X and Y. For a given query, we input the constraints in $\mathbf{K}$ as well as one additional set of constraints, which encodes the particular degree of freedom being tested. The degrees of freedom $X \rightarrow Y$, $X \leftarrow Y$, and $X \cdots Y$ are encoded by the sets of ASP constraints {`edge(X,Y).`}, {`edge(Y,X).`}, and {`-edge(X,Y). -edge(Y,X).`}, respectively. The hyphens (`-`) in the last set indicate negation to signify that neither edge is present between the nodes. In each run, the SAT solver returns either `SATISFIABLE` or `UNSATISFIABLE`, indicating whether the degree of freedom appears in at least one causal graph that is consistent with the constraints in $\mathbf{K}$. A system with N variables and three possible relations between each pair of variables will require $3\binom{N}{2}$ runs of the SAT solver to fully determine the degrees of freedom. Therefore, this procedure splits the set of all possible edge relations into two sets: (1) the degrees of freedom, each of which appears in at least one graph in the equivalence class, and (2) the relations that have been completely ruled out by the constraints. This procedure can be extended to consider cyclic causal graphs by including the degree of freedom indicated by the constraint set {`edge(X,Y). edge(Y,X).`}.

The degrees of freedom are used as the basis for our experiment-selection methods. We present two methods: the first is based on the degrees of freedom of the equivalence class; the second is based not only on the degrees of freedom but also an expectation metric. The first method is computationally less expensive because it does not require the enumeration of every causal graph in the equivalence class. The second method requires more computation, but its suggestions are correspondingly more informed, leading to more efficient causal discovery. Figure 3.16, adapted from [MWW17b], provides an overview of the proposed methods. Because of the

Algorithm 3.4.1 Deriving the degrees of freedom for an equivalence class

```
Data: $\mathbf{K}$: set of ASP-encoded causal-structure constraints over the set of variables $\mathbf{V}$
Result: $\mathrm{D}$: set of ASP constraints for the system's degrees of freedom
$\mathrm{D} \leftarrow \varnothing$
for each pair of variables $\{X,Y\} \in \mathbf{V}$ do
    for each set of constraints, $\mathbf{K}_d$, encoding a potential degree of freedom for $\{X,Y\}$ do
        $s \leftarrow$ satisfiability of constraint set $(\mathbf{K} \cup \mathbf{K}_d)$ if $s$ = SATISFIABLE then
            $\mathbf{D} \leftarrow (\mathbf{D} \cup \mathbf{K}_d)$
        end
    end
end
```

constraint-based causal discovery algorithm that we use, our approach can readily accommodate the background knowledge from a domain expert [Ebe17]. For instance, aside from the constraints obtained from statistical results reported in the literature, a domain expert may be able to articulate other causal-structure constraints that disallow direct edges between certain classes of variables, or that require certain paths involving specific subsets of variables. The ASP encoding that we employ can accommodate virtually any structural constraint that can be imposed on the edges of a causal graph.

Lastly, we present a method for categorizing hypotheses based on their utility for identifying a system's causal structure—a process that is usually infeasible to perform manually yet critical for conducting research efficiently.

3.4.1.1 Selecting experiments with degrees of freedom

Algorithm 3.4.2 gives an experiment-selection method based on the degrees of freedom. First, for each pair of variables in the system, $\{X,Y\}$, we obtain $n_{X,Y}$, the number of degrees of freedom in the equivalence class $\mathbf{E}$ for the pair $\{X,Y\}$, where $n_{X,Y} \leq 2$. Next, for the $(X,Y,n_{X,Y})$ three-tuple with the largest $n_{X,Y}$, we randomly choose one of the suggested experiments for the pair's degrees of freedom, $\mathbf{D}_{X,Y}$, as given in Table 3.3. (If multiple three tuples have the same maximum $n_{X,Y}$, we choose one randomly.) The experiments in Table 3.3 are chosen to be maximally informative, given the degrees of freedom that remain viable. For example, if the relations $X \rightarrow Y$ and $X \cdots Y$ are

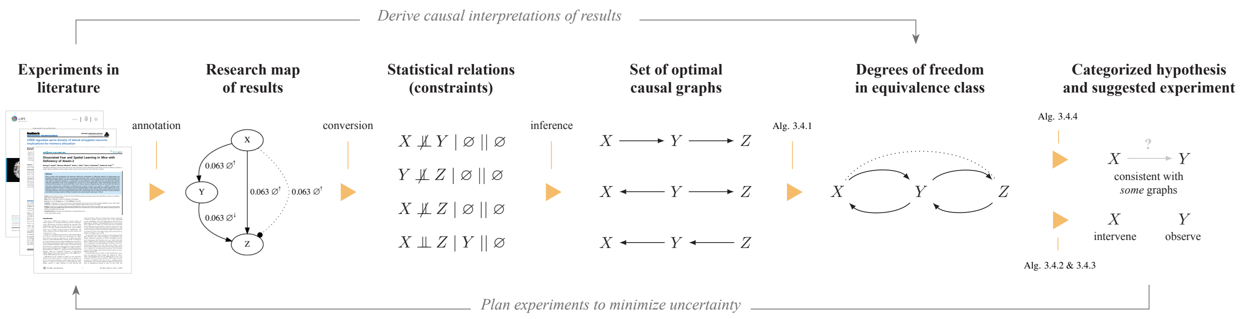

Derive causal interpretations of results
Experiments in literature
Research map of results
Statistical relations (constraints)
Set of optimal causal graphs
Degrees of freedom in equivalence class
Categorized hypothesis and suggested experiment
annotation
conversion
inference
Alg. 3.4.1
Alg. 3.4.4
X
Y
Z
0.063 ∅
$X \not\perp\!\!\!\perp Y \mid \varnothing \mid\mid \varnothing$
$Y \not\perp\!\!\!\perp Z \mid \varnothing \mid\mid \varnothing$
$X \not\perp\!\!\!\perp Z \mid \varnothing \mid\mid \varnothing$
$X \perp\!\!\!\perp Z \mid Y \mid\mid \varnothing$
$X \rightarrow Y \rightarrow Z$
$X \leftarrow Y \rightarrow Z$
$X \leftarrow Y \leftarrow Z$
?
consistent with some graphs
intervene
observe
Alg. 3.4.2 & 3.4.3
Plan experiments to minimize uncertainty

Figure 3.16: This block diagram provides an overview of the proposed method. Experimental results in the literature are annotated using the research-map schema; these results are converted into statistical relations in the form of ASP-encoded causal-structure constraints. An ASP-based causal discovery algorithm then computes the set of causal graphs that maximally accommodate the evidence. Algorithm 3.4.1 computes the degrees of freedom for the resulting equivalence class. Algorithm 3.4.2 and Algorithm 3.4.3 are used to identify informative experiments to perform next. Algorithm 3.4.4 categorizes hypotheses with respect to their utility for identifying a system's causal structure.

the remaining degrees of freedom, we do not suggest an intervention on Y, because intervening on Y would experimentally control the value of Y and thus preclude us from observing a correlation between X and Y that could arise if an $X \rightarrow Y$ relation were present in the true causal graph; intervening on Y effectively removes the $X \rightarrow Y$ edge, rendering the two degrees of freedom indistinguishable [Pea09]. The suggested experiments are therefore chosen for their ability to distinguish between the remaining degrees of freedom for a given pair of variables. Because this algorithm suggests an experiment given a set of experiments that have already been performed, additional bookkeeping is done to ensure that the experiments are not repeated unnecessarily (see the *while* loop in Algorithm 3.4.2). Within the *if* statement, the first condition ensures that if we have multiple competing sets of experiments, we choose the group of experiments that are least well represented in the set $\mathbf{P}$ (considering all the degrees of freedom, with a preference for the pair(s) of variables with the highest degrees of freedom). The second condition ensures that we choose an experiment from a pair of variables that has at least one experiment that has yet to be run. We enforce an explicit preference for experiments with variables that have not previously been selected. Note that in some edge cases, it is possible for our degrees-of-freedom approach to recommend only experiments that have already been performed. In these rare cases, we randomly choose an experiment that has yet to be run from the pool of all unperformed experiments.

Algorithm 3.4.2 Experiment selection based on degrees of freedom

```
Data: $\mathbf{K}$: set of ASP-encoded causal-structure constraints over the set of variables $\mathbf{V}$;
      $\mathbf{P}$: set of experiments performed to obtain $\mathbf{K}$
Result: $s$: experiment suggested on the basis of $\mathbf{K}$ and $\mathbf{P}$
$\mathbf{E} \leftarrow$ equivalence class (maximally) consistent with $\mathbf{K}$
$\mathbf{D} \leftarrow$ degrees of freedom for each pair of variables in $\mathbf{E}$ (Algorithm 3.4.1)
$\mathbf{R} \leftarrow \varnothing$
for each pair $\{X,Y\} \in \mathbf{V}$ do
    $n_{X,Y} \leftarrow$ number of degrees of freedom in $\mathbf{E}$ for $\{X,Y\}$  $\mathbf{R} \leftarrow \mathbf{R} \cup \{(X,Y,n_{X,Y})\}$
end
rank $\mathbf{R}$ by $n_{X,Y}$ in descending order  $c \leftarrow 0$  $m \leftarrow 1$  while $c < m$ do
    for each $(X,Y,n_{X,Y}) \in \mathbf{R}$ do
        $\mathbf{S}_{\mathbf{D}_{X,Y}} \leftarrow$ set of experiments suggested according to $\mathbf{D}_{X,Y}$ (Table 3.3)
        $m \leftarrow \max(\{m\} \cup \mid \mathbf{S}_{\mathbf{D}_{X,Y}} \mid)$
        if $\mid \mathbf{S}_{\mathbf{D}_{X,Y}} \cap \mathbf{P} \mid \leq c$ and $\mid \mathbf{S}_{\mathbf{D}_{X,Y}} \cap \mathbf{P} \mid < \mid \mathbf{S}_{\mathbf{D}_{X,Y}} \mid$ then
            $s \leftarrow s \in (\mathbf{S}_{\mathbf{D}_{X,Y}} - \mathbf{P})$
            return $s$
        end
    end
    $c \leftarrow c + 1$
end
return random experiment from set of possible experiments not in $\mathbf{P}$
```

3.4.1.2 Selecting experiments with degrees of freedom and expectation

When it is computationally feasible to compute every causal graph in the equivalence class, we can improve on the efficiency of Algorithm 3.4.2: Algorithm 3.4.3 gives an experiment-selection method that incorporates an expectation metric. As with Algorithm 3.4.2, this method uses the degrees of freedom of the equivalence class. But here the intuition is also grounded in expectation maximization. First, for each pair of variables in the system, $\{X,Y\}$, and for each possible degree of freedom, d, we obtain $m^d_{X,Y}$, the number of graphs in the equivalence class $\mathbf{E}$ that assign the degree of freedom d to the pair $\{X,Y\}$. We use this quantity to calculate the empirical probability of a graph in the equivalence class having that particular degree of freedom: $\frac{m^d_{X,Y}}{|\mathbf{E}|}$. We also calculate the number of graphs that would be eliminated from the equivalence class if we learned that this degree of freedom was the actual relation taken by that pair of variables in the true causal graph:

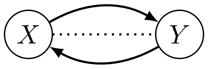
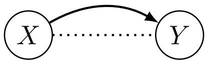
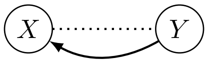
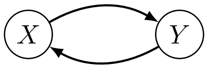
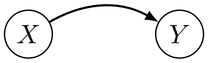
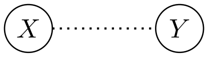
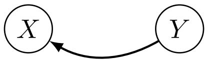
Degree-of-freedom pattern, $\mathrm{D}_{X,Y}$	**Suggested experiments,** $\mathrm{S}_{\mathbf{D}_{X,Y}}$
	$\mathbf{J} = \varnothing$ $\mathbf{J} = \{X\}$ $\mathbf{J} = \{Y\}$
	$\mathbf{J} = \varnothing$ $\mathbf{J} = \{X\}$
	$\mathbf{J} = \varnothing$ $\mathbf{J} = \{Y\}$
	$\mathbf{J} = \{X\}$ $\mathbf{J} = \{Y\}$
	$\mathbf{J} = \{X\}$
	$\mathbf{J} = \varnothing$
	$\mathbf{J} = \{Y\}$

Table 3.3: The experiments that would be most informative with respect to a pair of variables, given their particular degree-of-freedom pattern in an equivalence class. These suggested experiments inform the experiment-selection method given in Algorithms 3.4.2 and 3.4.3. The set $\mathbf{J}$ indicates which variables are intervened on in each experiment; when $\mathbf{J} = \varnothing$, a passive observation of the two variables is performed.

$|\mathbf{E}| - m^d_{X,Y}$. This empirical probability, $\frac{m^d_{X,Y}}{|\mathbf{E}|}$, is multiplied by its associated "reward," $|\mathbf{E}| - m^d_{X,Y}$, yielding the pair's expectation for a given d: $e^d_{X,Y} = \frac{m^d_{X,Y}}{|\mathbf{E}|}(|\mathbf{E}| - m^d_{X,Y})$. Next, for the $(X,Y,d,e^d_{X,Y})$ four-tuple with the highest expectation, we randomly choose one of the suggested experiments for d, as given in the last three rows of Table 3.3. (If multiple four-tuples have the same maximum $e^d_{X,Y}$, we choose one randomly.) As with Algorithm 3.4.2, additional bookkeeping is performed to ensure that experiments are not repeated unnecessarily.

3.4.1.3 Categorizing hypotheses by their utility for causal discovery

Given a knowledge base of constraints on causal structure, we define a method for placing a given hypothesis in one of three categories, with crucial distinctions:

1. *The hypothesis is consistent with* ***none*** *of the causal graphs in the equivalence class.* This kind of hypothesis should be pursued only if we are confident that one or more constraints in the current knowledge base are incorrect. The hypothesis is then useful insofar as it identifies which constraints in the knowledge base could be refuted. Otherwise, given the current knowledge base, we would fail to find even one causal graph that is consistent with this kind of hypothesis.

2. *The hypothesis is consistent with* ***all*** *the causal graphs in the equivalence class.* Although this kind of hypothesis produces accurate predictions about the system, it is equally unhelpful as the first kind with respect to experiment selection: this hypothesis should not be tested empirically unless we believe there to be a flaw in our current knowledge base and wish to refute one or more of its constraints. The reason is that if a hypothesis is consistent with *all* the causal graphs in the equivalence class, it already follows logically from the knowledge base; the logical proposition that expresses the hypothesis is thus true for all solutions (i.e., causal graphs). In propositional logic, it is said to be in the *backbone* of the satisfying formula [HHE13].

3. *The hypothesis is consistent with* ***some*** *(not all) of the causal graphs in the equivalence class.* This kind of hypothesis is most worth pursuing empirically. The experiment's result—which the current knowledge base cannot predict with certainty—is guaranteed to prune the equivalence class, bringing us closer to the true causal graph.

We categorize a hypothesis as follows: First, we express the hypothesis as a formal constraint that can be encoded in ASP. From Section 3.3.4 we see this can be achieved, for example, by adding a hypothetical edge to a research map of empirical results. Second, we query the SAT solver to see whether the hypothetical constraint is consistent with none, all, or some of the causal graphs in the equivalence class. As with the degree-of-freedom analysis, this procedure does not require the SAT solver to perform the expensive computation of enumerating every graph in the equivalence class. Instead, we can simply ask whether the hypothesized constraint is satisfiable, as a binary condition. If the answer is no, then we know that the hypothesis falls into the first category: it is consistent with none of the causal graphs in the equivalence class. If the answer is yes, then we must

Algorithm 3.4.3 Experiment selection based on degrees of freedom and expectation

```
Data: $\mathbf{K}$: set of ASP-encoded causal-structure constraints over the set of variables $\mathbf{V}$;
      $\mathbf{P}$: set of experiments performed to obtain $\mathbf{K}$
Result: $s$: experiment suggested on the basis of $\mathbf{K}$ and $\mathbf{P}$
$\mathbf{E} \leftarrow$ equivalence class (maximally) consistent with $\mathbf{K}$
$\mathbf{D} \leftarrow$ degrees of freedom for each pair of variables in $\mathbf{E}$ (Algorithm 3.4.1)
$\mathbf{R} \leftarrow \varnothing$
for each pair $\{X,Y\} \in \mathbf{V}$ do
    for each degree of freedom $d \in \mathbf{D}_{X,Y}$ do
        $m^d_{X,Y} \leftarrow$ number of graphs $\in \mathbf{E}$ with degree of freedom $d$ for $X,Y$
        $e^d_{X,Y} \leftarrow \frac{m^d_{X,Y}}{|\mathbf{E}|}(\mid \mathbf{E} \mid - m^d_{X,Y})$
        $\mathbf{R} \leftarrow \mathbf{R} \cup \{(X,Y,d,e^d_{X,Y})\}$
    end
end
rank $\mathbf{R}$ by $e^d_{X,Y}$ in descending order
$c \leftarrow 0$
$m \leftarrow 1$
while $c < m$ do
    for each $(X,Y,d,e^d_{X,Y}) \in \mathbf{R}$ do
        $\mathbf{S}_{\mathbf{D}_{X,Y}} \leftarrow$ set of experiments suggested according to $d$ (Table 3.3)
        $m \leftarrow \max(\{m\} \cup \mid \mathbf{S}_{\mathbf{D}_{X,Y}} \mid)$
        if $\mid \mathbf{S}_{\mathbf{D}_{X,Y}} \cap \mathbf{P} \mid \leq c$ and $\mid \mathbf{S}_{\mathbf{D}_{X,Y}} \cap \mathbf{P} \mid < \mid \mathbf{S}_{\mathbf{D}_{X,Y}} \mid$ then
            $s \leftarrow s \in (\mathbf{S}_{\mathbf{D}_{X,Y}} - \mathbf{P})$
            return $s$
        end
    end
    $c \leftarrow c + 1$
end
return random experiment from set of possible experiments not in $\mathbf{P}$
```

distinguish between whether the hypothesis is consistent with some or all of the graphs. We do this by querying for the satisfiability of the hypothesis's negation. If the hypothesis's negation *cannot* be satisfied by any of the graphs, then we know that the hypothesis falls into the second category: it is consistent with all causal graphs in the equivalence class. If the negation *can* be satisfied by at least one graph, then we know that the hypothesis falls into the third category: it is consistent with some (not all) of the causal graphs in the equivalence class. Therefore, any hypothesis, expressed

Algorithm 3.4.4 Hypothesis categorization based on logical satisfiability

```
Data: $\mathbf{K}$: set of ASP-encoded causal-structure constraints over the set of variables $\mathbf{V}$;
      $h$: ASP-encoded constraint that expresses a hypothesis
Result: $c$: categorization of hypothesis (category 1, 2, or 3 above)
$s \leftarrow$ satisfiability of $\mathbf{K} \cup \{h\}$ if $s =$ UNSATISFIABLE then
    $c \leftarrow 1$ return $c$
end
if $s =$ SATISFIABLE then
    $\hat{h} \leftarrow$ logical negation of $h$ $s' \leftarrow$ satisfiability of $\mathbf{K} \cup \{\hat{h}\}$ if $s' =$ UNSATISFIABLE then
        $c \leftarrow 2$ return $c$
    end
    if $s' =$ SATISFIABLE then
        $c \leftarrow 3$ return $c$
    end
end
```

as a causal-structure constraint, can be categorized with only one or two queries to the SAT solver (Algorithm 3.4.4). This categorization of hypotheses can guide experiment selection. Despite the enormous consequences that this categorization has on experiment planning, it is usually infeasible for a scientist to manually compute which category a hypothesis belongs to.

3.4.2 Evaluation

The experiment-selection policies given in Algorithms 3.4.2 and 3.4.3 were evaluated using the following simulation, which is given by Algorithm 3.4.5. First, one of the 543 possible DAGs over four variables was set as the true graph. Before any experiments were simulated, the equivalence class trivially contained every possible graph. To simulate how researchers learn about a system through repeated experimentation, we sampled study designs according to three different policies: at each iteration, we chose the next experiment (1) randomly, (2) according to Algorithm 3.4.2 (degrees of freedom), and (3) according to Algorithm 3.4.3 (expectation). The correct result of each experiment was returned by an oracle that assumed causal sufficiency and had access to the true causal graph. Each experiment's result was added to a growing list of constraints, yielding—at each iteration, and for each experiment-selection policy—an equivalence class of consistent causal graphs. After each experiment, we recorded the number of graphs that remained in each equivalence class. This process continued until we performed every one of the 48 two-variable studies defined

Algorithm 3.4.5 Evaluation of experiment-selection policies

```
Data: $\mathbf{G}_A$: all DAGs over $N$ variables;
      $\mathbf{P}_A$: all experiments over $N$ variables and their results, for each DAG $G \in \mathbf{G}_A$
Result: $\mathbf{S}_{\mathbf{P},G}$: sequences of experiments;
        $\mathbf{S}_{\mathbf{E},G}$: sequences of equivalence class sizes after each experiment
for each DAG $G \in \mathbf{G}_A$ do
    equivalence class $\mathbf{E} \leftarrow \mathbf{G}_A$
    set of performed experiments $\mathbf{P} \leftarrow \varnothing$
    while $|\mathbf{P}| < |\mathbf{P}_{A,G}|$ do
        $s \leftarrow$ experiment selected by policy (random, Algorithm 3.4.2, or Algorithm 3.4.3
        $\mathbf{P} \leftarrow \mathbf{P} \cup \{s\}$
        update $\mathbf{E}$ based on result of $s$ for $G$
        record $s$ in $\mathbf{S}_{\mathbf{P},G}$
        record $|\mathbf{E}|$ in $\mathbf{S}_{\mathbf{E},G}$
    end
end
compute average $\mathbf{S}_{\mathbf{E}}$ across every DAG $G \in \mathbf{G}_A$
```

by the research map schema. This simulation was repeated for every one of the 543 possible DAGs over four variables, thus showing that the experiment-selection policies are not sensitive to specific features of the true causal graph, such as the density of its edges. For each policy, we then computed the average number of graphs in the equivalence class that remained after each iteration (Fig. 3.17).

To show how our hypothesis-categorization method can inform experiment planning, we repeated the simulation in Algorithm 3.4.5 with an additional step: after each simulated experiment, we categorized the hypotheses implied by the remaining unperformed experiments and recorded the number of hypotheses that fell in each category. For instance, after 10 experiments were performed, 38 two-variable experiments remained to be chosen from, each implying its own hypothesis of independence (or dependence) between two of the variables in the system.[3] Given the knowledge base of constraints derived from the 10 performed experiments, we categorized each of the untested hypotheses and recorded the number of hypotheses that fell in each category. This process was

[3] For the simulation, each untested hypothesis assumed an independence relation; had we chosen to assume a dependence for each hypothesis, the counts for categories 1 and 2 would simply be exchanged. The effect of this choice is limited by averaging over all DAGs. What is most noteworthy is the proportion of hypotheses in category 3 to the proportion in either category 1 or 2.

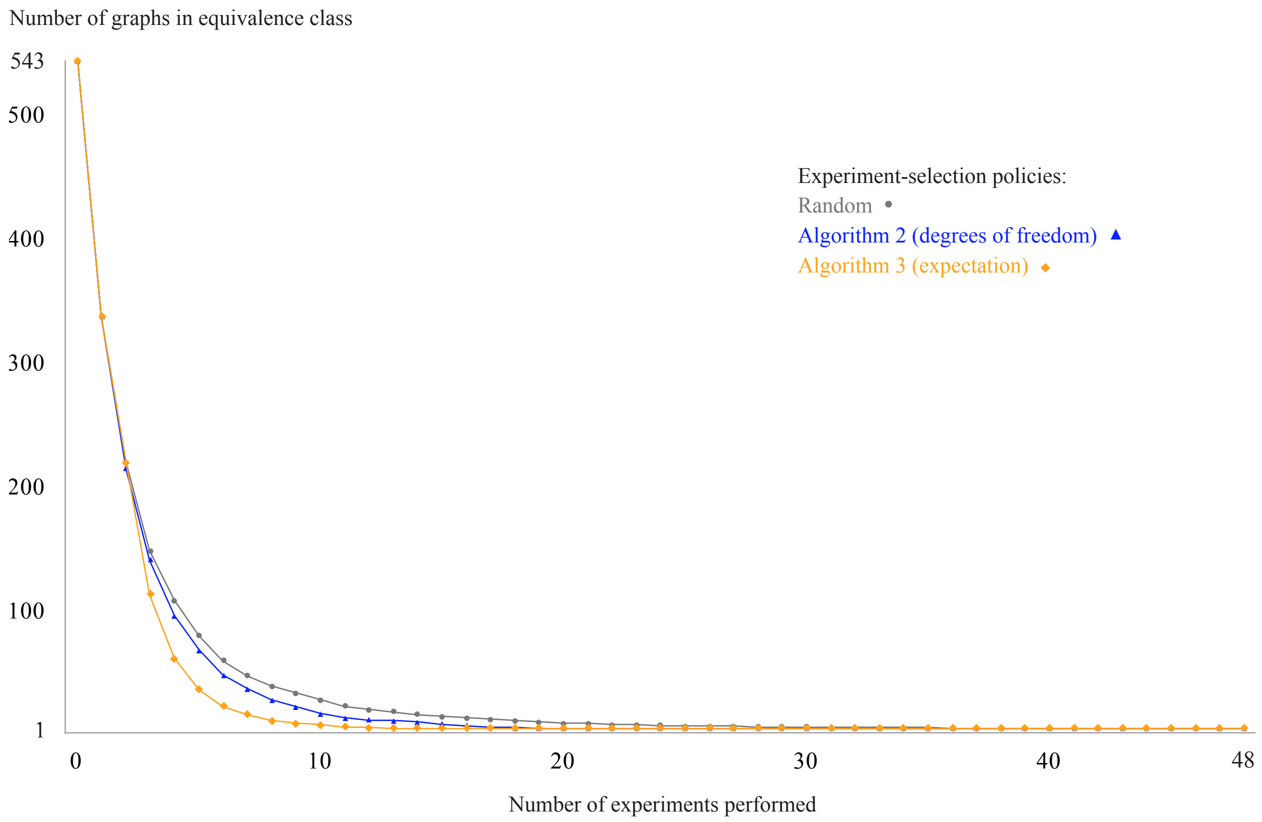


Figure 3.17: A comparison of three experiment-selection policies: (1) random, (2) Algorithm 3.4.2 (degrees of freedom), and (3) Algorithm 3.4.3 (expectation). This plot shows the results of the simulation given in Algorithm 3.4.5 for $N = 4$. The results show the experimental effort that is saved when each experiment is chosen based on the remaining degrees of freedom in the equivalence class.

repeated 543 times—once for each true DAG—and the counts of hypotheses in each category were averaged. The experiments were performed using an Intel Core i5-5250U x64 with 8 GB of RAM.

The results of the simulations given in Algorithm 3.4.5 show that selecting experiments strategically—that is, on the basis of the equivalence class's degrees of freedom—can save a considerable amount of effort in the laboratory: equivalent levels of underdetermination are reached with far fewer experiments using the suggestions of Algorithms 3.4.2 and 3.4.3 (Fig. 3.17). Table 3.4 shows the number of studies that each experiment-selection policy takes on average to reduce the equivalence class to various sizes. This table highlights that although Algorithm 3.4.2 and random selection require only one and two additional studies, respectively, to reach 50 graphs, they require far more studies to reach the minimum average number of graphs achieved by the simulation. Compared to the policy of Algorithm 3.4.3, the random policy on average takes 32 additional studies

	Number of studies needed to reach:		
Policy	**$<$ 50 graphs**	**$<$ 10 graphs**	**minimum**
Algorithm 3.4.3	5	9	15
Algorithm 3.4.2	6	14	23
Random selection	7	19	47

The number of studies that each experiment-selection policy takes on average to reduce the equivalence class to a given size.

Table 3.4: Empirical efficiency of experiment-selection policies

	Number of ASP models invoked for:		
Policy	**4 variables**	**8 variables**	**14 variables**
Algorithm 3.4.3	543	$\sim 10^{11}$	$\sim 10^{36}$
Algorithm 3.4.2	18	84	273
Random selection	0	0	0

The number of ASP models that each experiment-selection policy requires the solver to invoke in order to suggest an experiment.

Table 3.5: Computational efficiency of experiment-selection policies

to reach the minimum average value. Algorithm 3.4.3 reaches an equivalence class of fewer than 10 graphs—a reasonable number of graphs for a domain expert to review manually—in less than half the number of experiments required by the random policy (9 vs. 19).

As expected, Algorithm 3.4.3 (expectation) outperforms Algorithm 3.4.2 (degrees of freedom), but it does so at the cost of additional computation—a difference that can become quite significant for larger systems [HEJ14]. To give a sense of this difference, Table 3.5 shows the number of ASP models that each experiment-selection policy requires the solver to invoke before suggesting an experiment. Table 3.6 shows the average runtimes required to complete a single run of the simulations (i.e., for a given true causal graph) presented in Fig. 3.17 and Fig. 3.18, respectively, for each of the three experiment-selection procedures. Note that these runtimes reflect the interplay between the speed of each experiment-selection procedure and the additional computation required to consider varying numbers of causal graphs at each simulation step, as the procedures each reduced the equivalence classes at different rates. Comparing the runtimes for Algorithm 3.4.2 (degrees of

	Average execution time (s) to determine:	
Policy	**Graphs/equivalence class**	**Hypotheses/category**
Algorithm 3.4.3	61.7	246.5
Algorithm 3.4.2	34.0	527.7
Random selection	827.3	1001.5

The average runtimes required to complete a single run (i.e., for a given true causal graph) of the simulations presented in Fig. 3.17 and Fig. 3.18, respectively, for each of the three experiment selection procedures.

Table 3.6: Runtimes for experiment-selection and hypothesis-categorization simulations

freedom) and Algorithm 3.4.3 (expectation) to the runtime of random selection demonstrates that it is worth spending the extra computation time to identify the most informative experiments, in that far less computation is therefore needed to derive subsequent equivalence classes, which are appreciably smaller at each step given the informative experiment that is performed.

Fig. 3.18 presents the results of our hypothesis-categorization method's evaluation, which consist of the averaged counts of hypotheses in each category; Algorithm 3.4.2 was used as the experiment-selection procedure in the particular run that is displayed. On average, an appreciable percentage of the hypotheses fall into categories 1 and 2, which are far less informative than category 3 with respect to the goal of identifying a system's causal structure. As additional empirical results are added to the knowledge base—and the causal structure of the system becomes increasingly determined—the proportion of category-3 hypotheses becomes smaller. In other words, as we learn more about the system, it becomes harder to find informative hypotheses, and easier to make experimental predictions. This is to be expected, as the growing body of empirical results increases our knowledge of the system's causal structure. A scientist who wishes to determine a system's causal structure must therefore search for category-3 hypotheses—those represented by the "some" data series in Fig. 3.18—which is far more feasible using the hypothesis-categorization method presented above. Note that the runtimes for the hypothesis-categorization simulation (Table 3.6) reflect the time needed to categorize *every* untested, two-variable hypothesis for each of the 48 simulation steps. In real-world applications of the approach, scientists who are deciding whether to pursue a few different hypotheses could obtain their categories in less time.

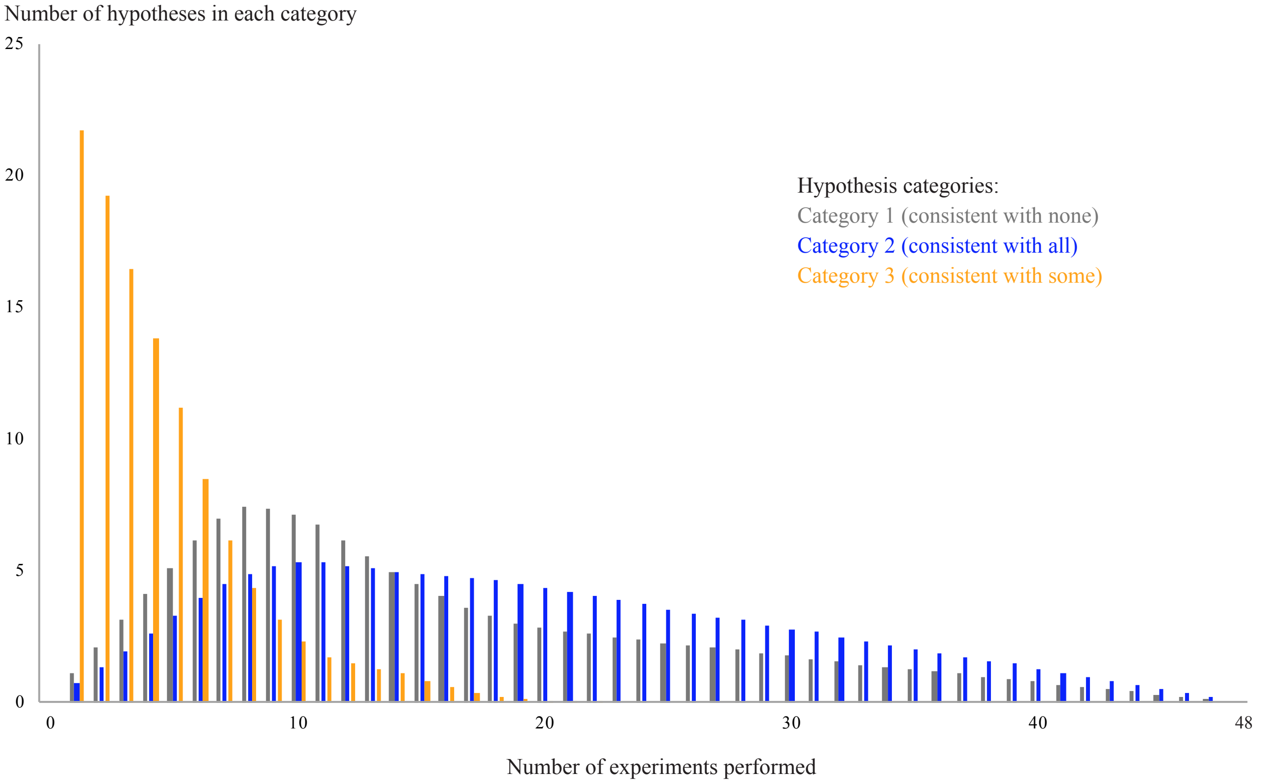


Figure 3.18: The average number of hypotheses that fell into categories 1, 2, and 3 in a run of the simulation given in Algorithm 3.4.5, in which Algorithm 3.4.2 was used as the experiment-selection procedure. As each experiment's result updates the knowledge base of causal-structure constraints, untested hypotheses may change categories, with important implications for the selection of the next experiment.

3.4.3 Discussion

The experiment-selection algorithms that we present are grounded in the type of graphical representation that many scientists—particularly biologists—already use to express causal mechanisms [LHM09]. As a result, scientists can readily interpret the algorithms' rationale for suggested experiments in the context of the graphical models that they consider to be viable. Although any experiment, if executed properly, can yield useful information regarding a system, strategic experiment selection—even if guided simply by heuristics—can save considerable amounts of work toward identifying a system's true causal structure. These savings are quantified by the simulations comparing Algorithms 3.4.2 and 3.4.3 to random experiment selection. Scientists who are constrained to piecemeal causal discovery can thus use these experiment-selection policies to avoid redundant experiments and select instructive ones, examining the degrees of freedom after each experiment to explore the range of edge relations that remain viable. After each empirical result is added to the knowledge base, the suggested experiments should be evaluated with respect to the full diversity of constraints on experiment planning that currently only a human being can consider, including technological limits, research funding, laboratory resources, and investigators' interests.

The comparison of Algorithms 3.4.2 and 3.4.3 to random selection does not imply that scientists are currently selecting their experiments at random. Instead, random experiment selection is used to establish a baseline of performance against which other methods can be judged; this approach has precedent in the experiment-selection literature [VJM00, Vat01, KWJ04, VJB06a]. Although scientists do not perform their experiments randomly, scientists in most fields do not plan their experiments in perfect coordination. These simulations thus highlight the experimental effort that can be saved when human experiment planning is more globally coordinated and augmented by computational tools—e.g., the ResearchMaps web application [MWD18] from Section 3.3—which formalize knowledge in a way that allows for automated inference.

Given that the number of causal graphs grows super-exponentially in the number of system variables, it is impractical to perform for larger systems the exhaustive simulations that we present here. Nonetheless, it is instructive to present the exhaustive simulations performed across all possible true graphs with four variables, as the dynamics of experiment selection can vary tremendously depending on the true graph; the simulations thus show our methods' average performance across all possible cases.

This approach is particularly helpful given that the limitations and optimal strategies for piecemeal causal discovery have been less well studied compared to the experiment-selection strategies for the general causal-discovery setting, in which it is assumed that every variable in the system can be measured in each experiment. In the context of causal discovery, our simulations thus allow for detailed analyses of the limitations of two-variable experiments, which are ubiquitous in the biological literature.

The presented experiment-selection procedures are still beneficial for a variety of real-world research settings. Although scientists regularly study large systems—with hundreds, thousands, or even millions of variables—experiments are often planned to identify relations between small subsets of variables; this is often true, for instance, in molecular and cellular neuroscience: researchers interested in the enormously complex system of the brain will choose to focus on a relatively small number of substructures to understand a particular neural circuit. Our approach can thus be applied iteratively on manageable subsets of variables, allowing researchers to "stitch" together findings to yield new inferences.

For example, in [MWW17b] we demonstrate how our degrees-of-freedom method can be used to combine the evidence in two neuroscience articles involving partially overlapping subsets of seven variables. Merging the results of the individual articles yielded a new inference regarding two variables that did not appear together in any of the experiments from the two articles; the resulting inference was deemed plausible by a domain expert.

For large systems, our methods can still render useful results when scientists can afford to wait relatively long amounts of time for supercomputers to return a solution [Ebe17]. Given that many experiments in science are very costly, taking months or even years to complete, experiment-selection methods that can save scientists multiple experiments toward identifying a system's causal structure can still be valuable even if they take days, weeks, or even months to return a result. For yet larger systems that fully exceed the scalability of our experiment-selection methods, researchers could still use our hypothesis-categorization method to evaluate whether a proposed experiment can further determine a system's causal structure, given a knowledge base of experimental results. Without having to enumerate every graph in the equivalence class, this approach can guarantee whether a proposed experiment will yield information that would reduce the number of viable graphs in the equivalence class.

As we demonstrate in [MWW17b], causal-structure information can be latent in the literature, yielding new inferences only when the right combination of findings are merged analytically. Such combinations may be difficult to find, making it impractical for a scientist to know with certainty whether a proposed experiment would yield information that is not already latent in the literature. Thus, if Algorithm 3.4.4 categorizes a proposed hypothesis in either the *none* or *all* categories, scientists can know with certainty that their existing evidence is sufficient to specify the outcome of the experiment that would test the proposed hypothesis.

The results of our simulations illustrate a few key points about the limitations of piecemeal causal discovery and the importance of planning experiments in light of the causal explanations that remain viable. It is known that $\log(N)+1$ experiments suffice to identify the true, causally sufficient DAG over N variables, where in each experiment, scientists can observe every variable in the system, and intervene on any number of variables in the system. If we are limited to single-intervention experiments, $N-1$ experiments are sufficient and in the worst-case necessary [EGS06, HEH13]. Under these assumptions, $\log(4)+1 = 4-1 = 3$ experiments suffice to identify the true DAG over the four variables considered in our simulations. But the experimental context we consider here is further constrained: we consider studies in which only two variables are observed simultaneously and at most one variable can be intervened on per experiment. Thus, on average, between four and five graphs remain in the equivalence class after every possible two-variable experiment has been performed. Our policies' inability to uniquely identify some of the true causal graphs is in part a manifestation of the limits on piecemeal causal discovery [Ebe13, May14, May19]. In future work, it would be instructive to better characterize how the efficiency of causal discovery improves as larger subsets of the system can be observed and intervened on simultaneously. Understanding exactly how much information is lost due to piecemeal causal discovery could help scientists to prioritize the development of laboratory equipment, including technologies that would allow for simultaneous observation of, and intervention on, larger sets of phenomena.

3.5 Piecemeal causal pipeline

Having introduced the details of individual components that can be connected together to facilitate knowledge discovery, we turn our focus into the formalization of this pipeline. The pipeline starts with a large corpus of scientific research papers and ends with discoveries made about individual causal components. Each stage is discussed in the context of an exhaustive dataset and novel improvements are made where applicable. This pipeline when pieced together can potentially become the "mind" inside the robot scientist.

Imagine the possibilities of a large collection of robot scientists, working in coordination day and night, each conducting experiments that will yield the maximal amount of information. In this setting, we may be able to exponentially increase the rate of scientific discovery. Although this vision may sound a bit fantastical, technologies and computational approaches are being developed that bring us closer to this goal. Indeed, scientists are already working alongside robot scientists: laboratory robots that can automatically conduct biological experiments and report their findings. However, robot scientists currently lack a “mind.”

This “mind” must have the ability to acquire all previous knowledge in order to make informed decisions. Of course, it is not trivial to acquire these disparate fragments of evidence representing all established knowledge within a field; it may be infeasible to do so. Instead, we focus on a type of scientific communication that contains much of the knowledge: scientific research articles. We believe that the most likely artifacts of scientists’ discoveries are published in such documents. Even inside this subset of knowledge, it is incredibly difficult to obtain and synthesize all the articles, some of which do not allow public access and thus might require significant financial resources. To this effect, we narrow our scope further and focus on the set of articles contained in the entirety of PubMed—a vast collection in order of 10^6 research papers and abstracts. In our study, the PubMed article collection becomes the artificial set of all known knowledge. Our robot-scientist “mind” is applied to the PubMed dataset, but its applicability is not limited to PubMed, nor are its methods tailored to this particular corpus. The techniques we present are applicable to any form of electronic publication—one in which we can easily convert the document to text.

Once we have a large set of existing knowledge (such as PubMed), we must synthesize the information into a representative form. This form must reduce the amount of information, taking in only what is necessary, while facilitating the discovery of contextual information. We intentionally focus on the domain of the biological researcher, because, as we detail in Section 3.3, discoveries can be represented in such a distinct form. Biological experiments commonly involve only two variables, due to the difficulty of obtaining simultaneous measurements in complex, living organisms, as well as such experiments’ technical difficulty and cost. A common way to extract meaningful biological phenomena described in text is to isolate the entities involved in experiments—typically an *agent* and a *target*. Often the text describes how a study—either an active intervention or passive observation (non-intervention)—provides evidence for an agent’s effect on a target. Thus, biological knowledge extraction typically involves some syntactical restrictions around the set: $\{$intervention, agent, target, effect$\}$. This setting leads naturally to the representation of

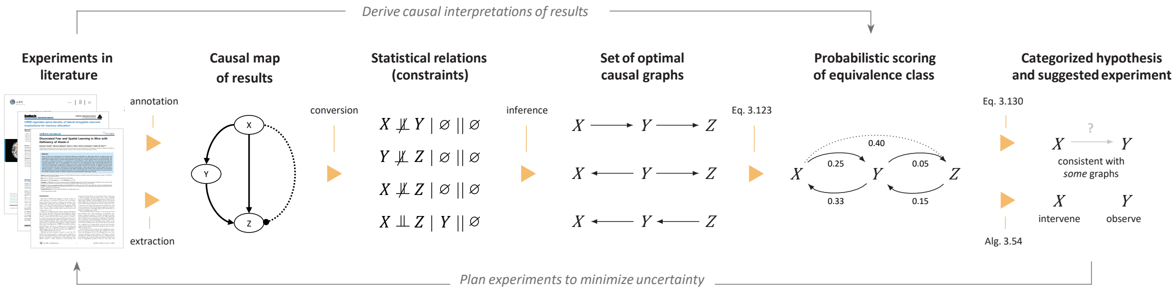

Derive causal interpretations of results
Experiments in literature
annotation
extraction
Causal map of results
X
Y
Z
conversion
Statistical relations (constraints)
$X \not\perp\!\!\!\perp Y \mid \varnothing \mid\mid \varnothing$
$Y \not\perp\!\!\!\perp Z \mid \varnothing \mid\mid \varnothing$
$X \not\perp\!\!\!\perp Z \mid \varnothing \mid\mid \varnothing$
$X \perp\!\!\!\perp Z \mid Y \mid\mid \varnothing$
inference
Set of optimal causal graphs
$X \rightarrow Y \rightarrow Z$
$X \leftarrow Y \rightarrow Z$
$X \leftarrow Y \leftarrow Z$
Eq. 3.123
Probabilistic scoring of equivalence class
0.40
0.25
0.05
0.33
0.15
Eq. 3.130
Alg. 3.54
Categorized hypothesis and suggested experiment
?
consistent with *some* graphs
intervene
observe
Plan experiments to minimize uncertainty

Figure 3.19: An updated pipeline (Figure 3.16) for the synthesis of causal text corpora into the selection of future experiments.

variables and relationships using a graph, where each node represents a biological entity, and each edge represents a relationship between entities.

With the semantics of representational knowledge established, we focus on determining which next experiment will yield the most information. To discover this informative experiment, we must first define a metric for information. For context consider the relationship between an agent and a target: If an agent and target are related, we can use the edge connecting their corresponding variables in a graph to specify their type of relation. For example, a target may increase whenever a connected agent is experimentally increased. In this case, we would say the entities have an *excitatory* relation. However, different researchers may report different conclusions for the same agent–target pair. Therefore, we must consider the true nature of an agent–target relationship to be a latent distribution over the class of all possible relations. Each reported finding is then a sampling from this distribution. Thus, we take information to represent an inferential calculation of this hidden variable at a given time x. In this model, every known agent and every known target are connected, with most connections having an information score equivalent to that of an uninformative prior. If we take the best experiment to perform next to be the one whose expected output yields the largest information gain, in the context of a hidden discrete distribution, this can be the experiment that causes the largest summation of each agent-target distribution divergence before and after the experiment.

In searching the experiment with the highest information gain, we uncover other potential avenues for knowledge discovery. For example, it may also be interesting to scientists to discover a previous unestablished relationship. For these discoveries, we can use graph-mining techniques to recommend a highly likely related agent–target pairs—which may or may not be the highest information-yielding experiment. We also explore discovering observed effects due to confounding variables—situations in which two variables appear to be related, but when all confounding variables are removed, they do not exhibit a previously established effect.

Together, these pieces compose the “mind” of the robot scientist. This pipeline, visualized by Figure 3.19, starts with aggregating all known sources of information and ends with a recommended experiment to perform. The recommended experiment to perform would then theoretically be used as input into the robot scientist, which would then feed the results back to the pipeline. Then, the pipeline would update the information about the known world and use this updated information to make an even more informed experiment recommendation. This self-enforced loop can result in a more refined implementation of the scientific method. By using more context than the human mind can process, and by performing calculations faster than the bare mathematician, we can elevate not only the robot scientist but the human scientist as well.

3.5.1 Components

We explain in more detail each of the processes representing the pipeline that makes up the robot scientist’s “mind.” First, we discuss the PubMed dataset, its aggregation, and meta-analysis. Next, we describe each component of the pipeline from start to finish. For each component, we present experiments and findings using PubMed and other datasets as input. We also discuss additional research areas related to each method and some potential improvements that can be investigated in future work.

3.5.1.1 PubMed dataset

PubMed is a collection of articles and abstracts comprising various sources. These sources include the MEDLINE database, online books, and other life-science journals. The retrievable subset of textual data exists in PubMed Central (PMC). As of the time of this writing, the PMC repository comprises 6.9 million articles, mostly from 10,644 journals. However, due to the difficulty and time in obtaining the complete set, the realized numbers differ. The dataset was downloaded via the file transfer protocol (FTP) over several months using a modest internet download speed. The

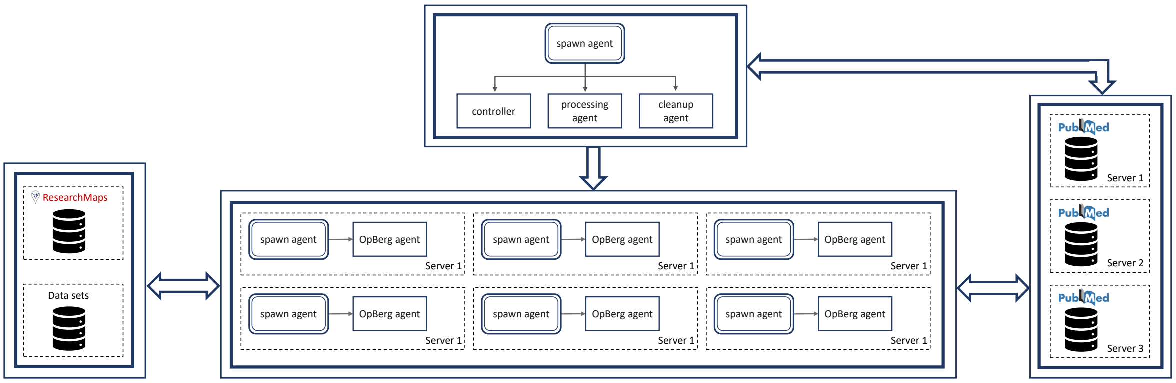


Figure 3.20: A visual representation of the process for causal extraction from the PubMed dataset.

FTP data was organized into partitions representing: older articles with text blobs obtained via optical character recognition, manuscripts in extensible markup language (xml) and raw text, bulk collections of journal articles in a unique scientific literature-based xml format (nxml) and raw text, where each article is packaged together with the original pdf, nxml extract, and images used in the articles—and lastly just the articles as their submitted pdf. There is some overlap between these described partitions; however, we obtained all the data described above for completeness.

3.5.1.2 Causal extraction

As we described in Section 3.2, OpBerg is an effective technique to extract causality when we do not have a large set of labeled input data, and when the candidate output set is large. The PubMed collection surely satisfies this latter requirement. The ResearchMaps database also provides a modest set of labeled data. However, each labeled data represents only one labeling for a particular way to describe causality. Thus, we have a minuscule set of labeled input data for each causal statement. The existing dataset represents the ideal conditions for OpBerg, and indeed the potential number of causal assertions can be significantly increased.

Another advantage of using OpBerg lies in the nature of comparing sequences. Given that we will match, mismatch, insert, or delete each character in an alignment, we can easily determine the constituent parts of the labeled input and match these with the found sequence. For instance, consider the following sentence:

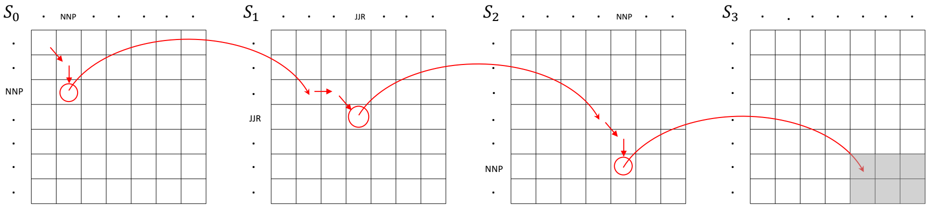
> *The preferential distribution of Arc$^+$ nuclei in neurons with higher CREB function was also observed immediately after training.*

After changing this sentence into the POS mapping of:

> DT-JJ-NN-IN-NNP-NNS-IN-NNS-IN-JJR-NN-NN-VBD-RB-VBN-RB-IN-NN

we can match the NNP (Arc$^+$), JJR (higher) and NN (CREB) in the labeled data with the candidate sentence. These matches allow us to build the corresponding biological representation.

To extract the labeled data's necessary parts requires OpBerg to match or mismatch on the same or similar POS tags in the candidate fragment. As the algorithm is given, this will not always be the case. Since it is discovering the optimal amount of breakpoints and then performing local alignments over those breakpoints, there is no guarantee that a labeled set of important POS tags will be matched or mismatched. In order to enforce these requirements, we must add additional matrices to hold states of where the current alignment is. Each time two selected POS tags are compared, we can then transition to the next state. The result then lies in the final comparison matrix. In the Arc$^+$ example in which Arc$^+$ leads to an increase in CREB, we would first start in an initial state S_0. Then, when we compare two NNP tags, we would transition to S_1, and after two JJR tags, we would go from S_1 to S_2, and finally, two NN tags would take us from S_2 to S_3. The final result lies in the max matrix cell of S_3.



We must add these additional matrices and recurrent relations as a function of the necessary sequence length N for sequence $C_1C_2 \ldots C_N$. The initial state (S_0) resembles the original equation form, while each additional state must be reflected in the original equations' matrices. The necessary addition to Equation 3.45–Equation 3.60 is given as: For state S_0

$$\delta(i,j)_{S_0} = Max \begin{Bmatrix} 0 \\ \\ L_{I,S_0}(i_p, j_p) + S(a_i,b_j) \\ \\ L_{G,S_0}(i_p, j_p) + S(a_i,b_j) \\ \\ L_{D,S_0}(i_p, j_p) + S(a_i,b_j) \end{Bmatrix} \tag{3.94}$$

and for each state u from 1 to 3:

$$\delta(i,j)_{S'_u} = Max \begin{Bmatrix} 0 \\ \\ L_{I,S_u}(i_p, j_p) + S(a_i,b_j) \\ \\ L_{G,S_u}(i_p, j_p) + S(a_i,b_j) \\ \\ L_{D,S_u}(i_p, j_p) + S(a_i,b_j) \end{Bmatrix} \tag{3.95}$$

$$\delta(i,j)_{S''_u} = Max \begin{Bmatrix} 0 \\ \\ L_{I,S_p}(i_p, j_p) + S(a_i,b_j) \\ \\ L_{G,S_p}(i_p, j_p) + S(a_i,b_j) \\ \\ L_{D,S_p}(i_p, j_p) + S(a_i,b_j) \end{Bmatrix} \tag{3.96}$$

$$\delta(i,j)_{S'''_u} = Max \begin{Bmatrix} \delta(i,j)_{S'_u} \\ \\ \delta(i,j)_{S''_u} \end{Bmatrix} \tag{3.97}$$

$$\delta(i,j)_{S_u} = \begin{cases} \delta(i,j)_{S'''_u} & \text{if } a_i = b_j \\ \delta(i,j)_{S'_u} & \text{otherwise} \end{cases} \tag{3.98}$$

We take the shorthand p to represent the previous item ($i_p = i - 1$); S is a scoring matrix for two POS tokens, and each L is a matrix representing a match/mismatch, insertion or deletion. More details are presented in Section 3.2.1. Under these new equations, we can find the OpBerg alignment that crosses through each sequence in C.

The other consideration OpBerg must make is the massive number of candidate sentences to consider. The expected running time is in the order of days, so any improvement can be quite impactful. We take the approach of running OpBerg concurrently. We can do this most simply by splitting the labeled input set between processes running on different clients. However, we still need synchronization for processing the PubMed data, such as uncompressing files or extracting textual data from a pdf. To avoid unnecessary disk usage, we keep many individual compressed files in their original form and extract them only during processing. When the uncompressed files are not in use, they are deleted to save space. To keep them uncompressed was determined to be infeasible given their collective size. We take the approach of having one agent in charge of processing each PubMed data item. After preparing the PubMed item for consumption, a number of agents running across a compute cluster then consume the formatted text. After consumption, the data item is marked ready for deletion, and a cleanup agent then removes the already processed items. To ensure synchronization and to handle any possible errors, a controller agent communicates to each process. In the event of an error, the controller will respawn the necessary agents. Each cluster actually hosts a spawn agent that creates a process that will run, and is also in communication with the controller agent. The entire process flow is given in Figure 3.20.

3.5.1.2.1 Experimental Setup

A six-compute cluster was set up to process the entirety of the PubMed dataset. In the first stage of processing, we start with the ResearchMaps dataset and retrieve the corresponding free text for each edge. In order to find the free text, we first find the matching PubMed data item. This is no easy task given the PubMed data item may not always have a title, or might not label a block of text as being the title. Additionally, the text from the ResearchMaps dataset may be different or contain spelling errors. For PubMed articles that were well-formed, such as some xml articles, we can easily take the section labeled as the title and perform a local alignment over the ResearchMaps title, which is well-formed but might contain spelling errors. Other PubMed articles are given as blobs of texts that do not easily identify the title. In these cases, we take the first half of the text blob and run a local alignment over the characters. We take this notion because we assume with very high probability the first half will both contain the title, and not contain the references section. We also take the assumption that the title is unique enough so that it is not commonly used in free text. Since we cannot guarantee this uniqueness, we end up keeping the top 10,000 highest scoring matches against each title.

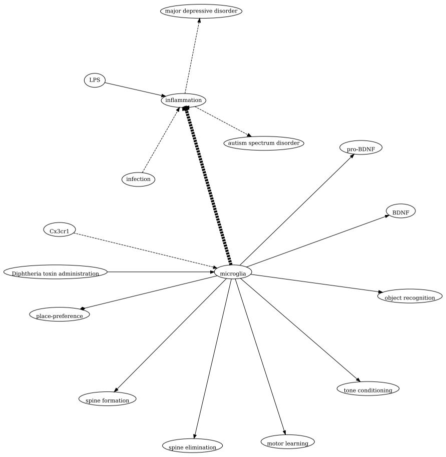


Figure 3.21: An extracted relation connecting two ResearchMaps fragments.

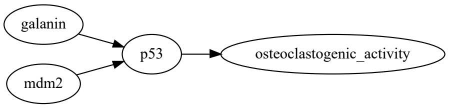


Figure 3.22: A piecemeal connected graph of extracted causal relations.

Another consideration must be made in comparing a ResearchMap title with a relatively small amount of characters to half a published PubMed article with a large amount of characters. With the expansive size of the PubMed dataset we must consider faster ways to make a comparison. To obtain this efficiency goal we consider comparing words rather than only characters—with the assumption that the words (W) compared will be significantly less than the number of characters (C) compared. To allow for spelling mistakes among the same word we tweak our scoring matrix to score matches as words that are similar to each other. We define similar as having a finite set of insertions, deletions, or mismatches in a global alignment. By limiting the number of non-matches we can compare two sequences both of size N in $\mathcal{O}(N)$ time. This will reduce our title matching running time from $\mathcal{O}(W^2C^2)$ to $\mathcal{O}(W^2C)$.

After the title matching, the next task is to extract the causal statements representing each research map. Since we do not know exactly where the map is represented in the text, we assume

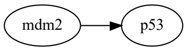
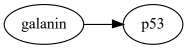
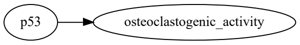
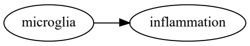
Sentence	Edge
*The inhibition of **Mdm2** by nutlin-3 increased the basal **p53** protein level and rescued its tetanization-induced depletion, which suggested the involvement of Mdm2 in the control over p53 during LTP [PLS15]*	
*In addition, **galanin** inhibited the Abeta(25-35)-induced dysregulation of **p53**, Bax, and MAP2 in rat hippocampus [CY10]*	
*On the other hand, **p53** deficient osteoblasts show enhanced **osteoclastogenic activity** compared to normal osteoclasts [WL07]*	
*The finding that the interactions between mutant SOD1s and chromogranin A&B leading to secreted mutant SOD1s that in turn can act on **microglia** to cause **inflammation** could be responsible for the non-cell autonomous effect of the toxicity [SCL14]*	

Table 3.7: The sentence, publication, and piecemeal causal graph fragment of extracted causal relations from the PubMed dataset.

that this assertion must be specified in a sentence. To this end, we split the body and abstract of the article into its constituent sentences using the established sentence parser in the Stanford CoreNLP package [Man14]. We then search for sentences that contain both the agent and target for an edge. If there exists more than one sentence, we submit these for manual review.

With our labeled data established, we seek to find similar POS sentences used throughout the PubMed dataset. We process each PubMed article in a pipeline fashion previously described using the computing cluster. Each labeled data is previously broken into its POS tags, again using the Stanford CoreNLP package, and compared against a text blob by taking the original articles and breaking them into sentences, then each sentence into a POS sequence of tokens. The tokens are compared using Equations 3.94–3.97. For each labeled input item, we record the top 10,000 matches for further analysis. Additionally, we record the top 10,000 matches for both raw character matches as well as word similarity matches computed by comparing embeddings.

After the entire PubMed dataset is searched, we refine the best 10,000 matches for each labeled input item. Then, we filter out candidates whose matching POS tokens to the agent and target are not in a database of known biological entities. This database was constructed using existing ontologies, such as the Neuroscience Information Framework (NIF) Standard Ontology [NIF], combined with a database of representations from Section 3.3 consisting of agents and targets. We then seek to find matches that have similar verb token alignments. We take similar to be within a threshold of distance between embeddings for a particular word.

3.5.1.2.2 Experimental Results

We show the results of some example sentences and their corresponding research map causal representation in Table 3.7. Additionally, we map all the found edges in the global causal graph shown in Figure 3.24(b). The results shown have the potential for impactful discoveries in the biological domain. As shown in Figure 3.22, we can suggest discoveries that are implied to be true. In this case, the figure shows that osteoclastogenic activity has an excitatory response to MDM2. We invite the biological researcher to evaluate this claim—that is, to design an experiment where a change in MDM2 is introduced experimentally or observed passively, and the effect on osteoclastogenic activity is observed. We can also use this approach to discover potential treatments that may not be stated explicitly in the literature. For example, by connecting microglia to inflammation, as in Figure 3.21, we can immediately deduce diphtheria toxic administration as a treatment candidate for autism spectrum disorder. Of course, these deductions must be scientifically validated, but any one of them could lead to meaningful treatments. Table 3.7 gives a brief list of such deductions, and we show a larger set of connections in Figure 3.23.

3.5.1.3 Graphical representation

With the causal declarations extracted, as shown in the previous section, we can now combine these with existing knowledge bases and seek to synthesize the information down into a form more suitable for knowledge discovery. For the task of mapping biological discoveries, we are somewhat limited to knowledge bases that have cataloged relations as described in Section 2.1.1.1. Nevertheless, some do exist, while others have been obtained through manual annotations. A more extensive set exists when we seek to catalog only causal relations. While not as expressive as the full biological relationship semantics, causality can still be tremendously valuable for experiment selection or recommendation tasks. This relaxed constraint allows us to include many other established datasets alongside the sets used for biological relationships. We describe the sets used in Table 3.8.

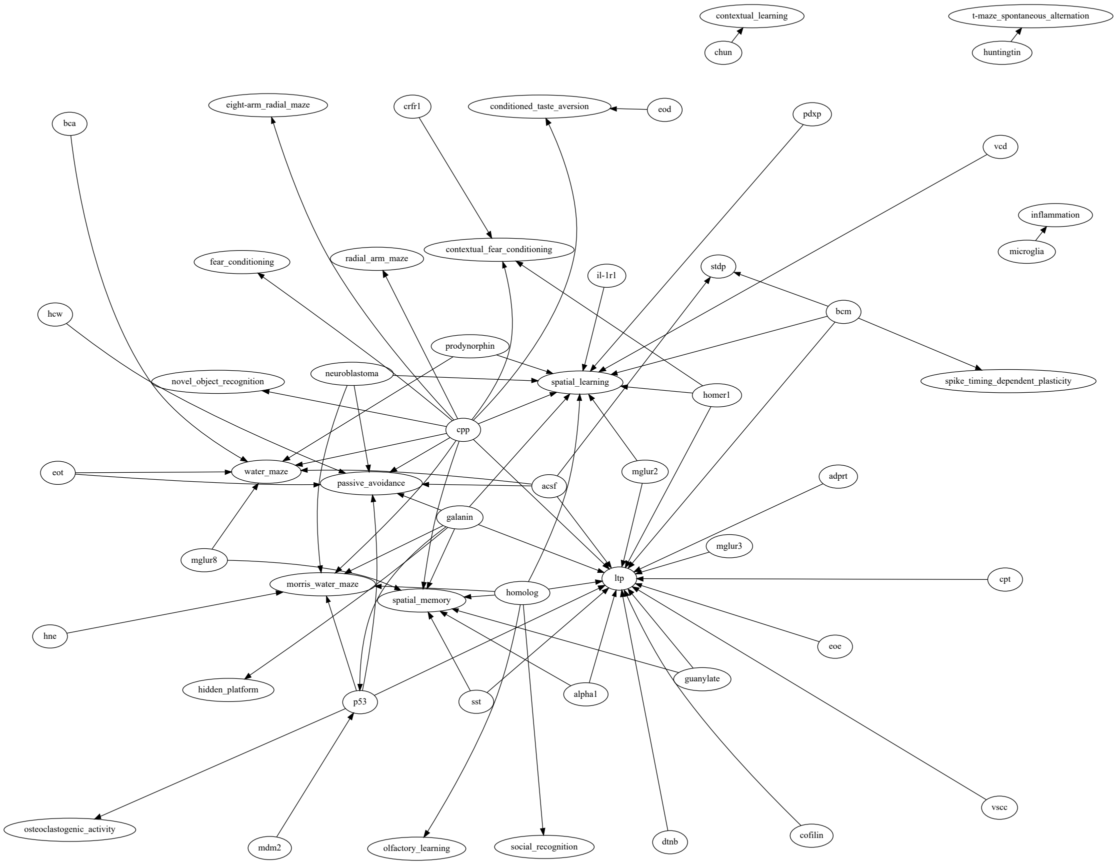


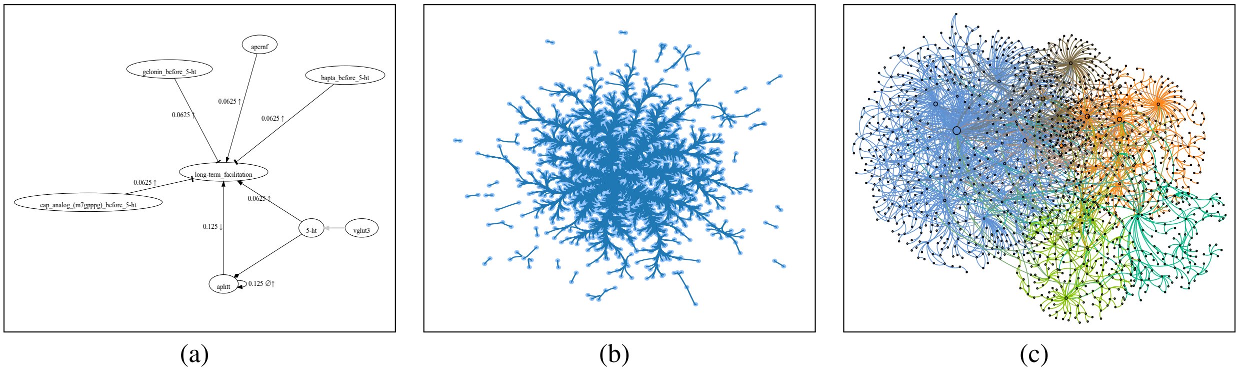
Figure 3.23: The causal network of a subset of relations discovered in the experiment described in Section 3.5.1.2.

Alongside the relationship type between an agent and target, we seek to express a measure about that relationship. For biological relationships, this was already established in Section 3.3.3—the main idea is that an experimental observation is just a sampling from a latent distribution over the possible relationships. For biological relationships, we assume a generative model of:

```
Choose $\theta_r \sim \text{Dir}(\alpha_r)$
for $c \in \{\uparrow, \varnothing^{\uparrow}, \varnothing^{\downarrow}, \downarrow\}$ do
    Choose $\theta_c \sim \text{Dir}(\alpha_c)$
end for
for $t \leftarrow 1$ to $\infty$ do
    Choose $c \sim \text{Multinomial}(\theta_r)$
    Choose $e_t \sim \text{Multinomial}(\theta_c)$
```

Name	Description	Classes	Sentences
RM46	ResearchMaps collection of 46 neuroscience articles	2,4,7	200
LLL05	Causal sentences and papers extracted from the LLL05 Challenge	2	131
NDE27	Non-domain experts labeling of various PubMed articles	2	1,025
BioCause	The biomedical discourse causality corpus and corresponding articles	2	1,000
RM6	Domain expert labeled set of articles from ResearchMaps	2,4,7	356
RM	The entire ResearchMaps database mapped	2,4,7	8,693
PUB	Extracted piecemeal causal statements from PubMed found in Section 3.5.1.2	2	713

Table 3.8: Datasets used in the graphical representation of the global causal graph.



(a)

(b)

(c)

Figure 3.24: A example causal map from the ResearchMaps database (a) alongside the global graph—a graphical representation of the entirety of our aggregated piecemeal causal information (b) and a subset the global graph showing clearly formed communities (c).

```
8: end for
```

From the observations, we score the edge as the maximum posterior probability over all classes and relations—an expression of the concepts known as *consistency* and *convergence*. *Consistency* refers to the obtaining of evidence for a particular relation using repeated iterations of the *same* experimental method. *Convergence* refers to the obtaining of evidence of a particular relation using *different* methods. This approach thus mirrors the epistemic principles, detailed in Section 3.3, that biologists use to evaluate the strength of the evidence for a relationship.

We seek to develop a similar scoring approach for causal-only biological relationships (instead of the more elaborate taxonomy used in research maps). For each biological entity pair A and B, we take the above generative model as a guide and assume a latent distribution over two classes: $A \rightarrow B$ and $A \perp\!\!\!\perp B$. Each experiment that implies a causal relationship is then just a draw from this distribution. For shorthand, we will represent these classes as: $\rightarrow$ and $\perp\!\!\!\perp$ respectively with a third notation, $\leftarrow$, fully expressed as $A \leftarrow B$, equivalent to $B \rightarrow A$.

We reiterate that we limit the scope of relationships to include only pairs of variables—an ubiquitous experimental setting in much of the biological sciences, given the cost and complexity of such experiments. Likewise, we reiterate that we are assuming causality has already been implied. Our method thus picks up where rigorous statistical and numerical analysis is already assumed to have been performed, and the result is published.

The causal-only generative model becomes much more straightforward than the fully expressive biological-relationship model for the set of relations, however to accommodate for a more general set of input data we add an additional experiment class representing an unknown intervention ($\varnothing$). The generative model thus takes the form of Equation 3.74–Equation 3.78 with C and R as $\left\{\uparrow, \varnothing^{\uparrow}, \varnothing^{\downarrow}, \downarrow, \varnothing\right\}$ and $\{\rightarrow, \perp\!\!\!\perp\}$ respectively. With the causal class substitutions of C and R we can arrive at the causal equivalent of Equation 3.86, referred to as the causal-only piecemeal causal evidence index (CPCEI).

With the scoring method set between a relationship, we seek to represent relationships in a meaningful way. ResearchMaps created the foundation for this work regarding biological relationships by representing a biological finding as an agent acting (or not acting) on a target. This representation takes the form of a graph where biological entities are nodes and their relationships are edges. A score on the edges represents a measure of the strength of the relationship. A visual example is given in Figure 3.24(a). Each edge type represents the highest-scoring categorical relationship denoted as a conclusion, and the edge identifies the type of experiment performed. Note that an agent and target can have multiple types of experiments performed between them. In this case, we append the type of experiment onto the edge.

We seek an analogous approach for describing causal relationships. We model each agent and target as nodes and edges, denoting the maximally likely causal class. At time 0, we assume a non-informative prior and take $\perp\!\!\!\perp$ as the assigned class. We choose a no-dependence class as the initial class because we believe this most accurately represents the relationship between two randomly chosen biological entities. When stitched together, the set of all biological entities and their relationships form a massively connected graph. We can then use data-mining and graph-analysis techniques to discover knowledge about the known biological world. Since we draw an edge between an agent and target that is initialized as $\perp\!\!\!\perp$, the graph becomes a complete graph—though we can easily change the edge types to suit whatever data-mining task we are performing, such as assuming the absence of an edge represents a $\perp\!\!\!\perp$ edge.

3.5.1.4 Experimental Setup

To show our complete known biological causal world, we synthesize all of our causal biological data sources (Table 3.8) into our graph representation described above. We leave out the no-connection edge type to illustrate the complex nature of causal biological networks.

3.5.1.5 Experimental Results

Figure 3.24(b) shows the entirety of our collected sources as a causal network. As we can see, this world is quite complex. It is interesting to see how many variables are related to each other and the scope of the documented biological world. Alongside the complete causal network, we highlight an interesting find about communities. Using modularity [BGL08, LDB08], we can see that there does exist distinct communities that involve many different biological entities, shown in Figure 3.24(c). As we show later in our discussion on experiment recommendation, these networks can help guide which experiments to conduct and may be helpful in other knowledge discoveries as well.

3.5.1.6 Experiment Selection

Given a causal graph with nodes representing biological entities, and edges representing relationships among these entities, we seek to develop an approach to acting upon this representation in an experimental setting. We presume it is the biologist's implicit desire to perform an experiment that yields the most information. But what does yielding the most information translate to in the setting of a causal graph? We submit that this concept entails determining the true causal structure of the graph as completely as possible. For the purpose of intuition and simplicity, we begin this discussion by making a few simplifying assumptions about the causal world.

As described in the previous section, when any biological element, A, influences another element, B, we model this effect as $A \to B$. This implies a causal relationship of A having a causal effect on B. A natural extension to the graphical representation is a directed acyclic graph (DAG), with an edge departing from the A node and directed to the B node. We also trivially model the absence of a direct causal relationship between A and B as the absence of an edge connecting the two nodes ($\perp\!\!\!\perp$ edge). Furthermore, we can simplify the DAG representation by assuming constraints of piecemeal causality. One example constraint would be: if $A \rightarrow B$ and $B \rightarrow C$ then $A \rightarrow C$ or alternatively expressed as $A \not\perp\!\!\!\perp C \mid B \,||\, \varnothing$. A more detailed discussion on piecemeal causal constraints is given in [MWW17b, Mat17, MWW21]. Considering the transitive closure for our set of edges (E) as E^{+}, the simplified representation of our graph can be formalized as:

$$C_1(a,t,E) = \begin{cases} 1 & \text{if } a \neq t \land (t,a) \notin E^{+} \\ 0 & \text{otherwise} \end{cases} \tag{3.99}$$

$$C_2(a,t,E) = \begin{cases} 1 & \text{if } (d_i,t) \in E^{+} \; \forall d_i \in \{(d,a) \mid (d,a) \in E^{+}\} \\ 0 & \text{otherwise} \end{cases} \tag{3.100}$$

$$C_x(a,t,E) = \begin{cases} 1 & \text{if } C_1(a,t,E) = 1 \land C_2(a,t,E) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.101}$$

$$V = \{\text{all biological entities}\} \tag{3.102}$$

$$G = (V,E) \tag{3.103}$$

$$E \subseteq \{\{a,t\} \mid a,t \in V^2 \land C_x(a,t,E) = 1\} \tag{3.104}$$

And a formal desideratum of: *Given a set of vertices V of size z and possible edges E_z^*, the scientist is to determine the true casual graph G_z'.*

In our theoretical setting, the discovery of G_z' is a sequential process whereby each experiment is conducted at time i which adds the agent and target pair, (a_i, t_i), subjected to the existing causal constraints, to E, the set of experiments performed so far. Each experiment's outcome is considered an observation from a hidden variable, and the result allows us to construct all possible causal graphs consistent with the information at time i. Alternatively, we can view this as eliminating a set

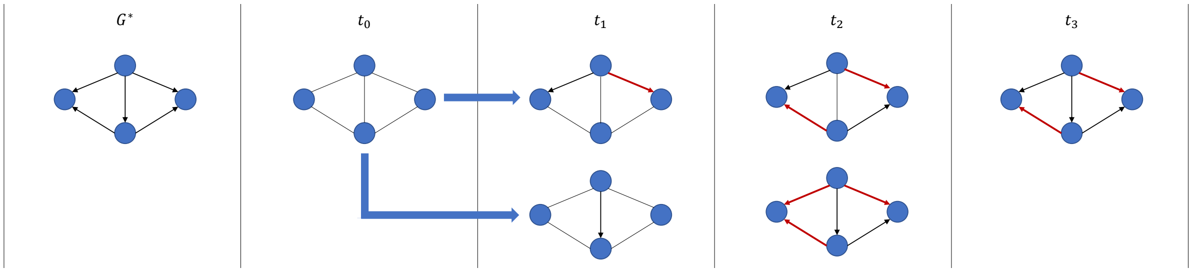


Figure 3.25: An example of different experiment selections leading to different number of experiments needed to find G'.

of graphs from the existing possible graph set at time i. Discovering the true graph G'_z is equivalent to eliminating all but one graph from the set of all possible DAGs of size z, G^*_z. Using the shorthand for the powerset as $\mathcal{P}$ we write this more formally:

$$E^*_0 = \{\Delta \in \{\mathcal{P}(V) \times \mathcal{P}(V)\} \mid C_x(a{,}t{,}\Delta) = 1 \ \forall \{a,t\} \in \Delta\} \tag{3.105}$$

$$E^*_n = \{\Delta \in E^*_{n-1} \mid \{a_n, t_n\} \in \Delta\} \tag{3.106}$$

$$E' = \{\{a,t\} \in \Delta \mid \Delta \in E^*_k \wedge |E^*_k| = 1\} \tag{3.107}$$

$$G' = (V, E') \tag{3.108}$$

For simplicity, we assume the biologists to operate under a naive model whereby each experiment is chosen randomly. We assume a generative model for the process a scientist takes for causal graph discovery as:

```
$E^* \leftarrow$ Set of all possible edges
while $|E^*| > 0$ do
    Choose $e_i \sim$ Uniform$(E^*)$
    $r \leftarrow$ Bernoulli$(p_{e_i})$
    if $r = 1$ then
        $E^- \leftarrow \{\Delta \in E^* \mid e_i \in \Delta\}$
    else
        $E^- \leftarrow \{\Delta \in E^* \mid e_i \notin \Delta\}$
    end if
    $E^* \leftarrow E^* \setminus E^-$
end while
```

Given this random science model (Random), we seek improvement by solving a simple problem: discovering the casual graph G'_z from E^*_z such that the number of experiments performed, k, is minimized. We denote this as E'' defined as:

$$E'' = \min_{\forall k \in \mathbb{W}} \{\{a, t\} \in \Delta \mid \Delta \in E^*_k \wedge |E^*_k| = 1\} \tag{3.109}$$

The optimal algorithm thus chooses the smallest possible k, which leads to G'_z, or equivalently the smallest set of experiments (e'') that lead to E''. But what is the minimal k? Although this is not an easy question, we prove k to be bound by the following constraint: $|V| - 1 \leq k \leq |V| \cdot (|V| - 1)$.

Theorem 3.5.1. $|V| - 1 \leq k \leq |V| \cdot (|V| - 1)$

Lemma 3.5.2. $k \leq |V| \cdot (|V| - 1)$

Proof. Given each edge e corresponds to a casual relationship, and each experiment x either implies or denies a causal relationship of the form $a \rightarrow t$, we either eliminate $\{\Delta \in E \mid e \notin \Delta\}$ if the experiment is true or $\{\Delta \in E \mid e \in \Delta\}$ if the experiment is false. Since the total edge directionality is 3, this implies in the worst case 2 experiments, since after 2 experiments, the total size is 1 and thus the result can be inferred. □

Lemma 3.5.3. $|V| - 1 \leq k$

Proof. Proof by induction:

Base case. Let $(v_1, v_2) = V$, then there must be at least one experiment conducted since no result can be inferred a priori.

Induction step. Given G'_z for which we assume $z - 1 \leq k$. Let v_{z+1} be a newly added node to G'_z, if we represent G'_z as a node, v_G, where $(v_{z+1}, v_G, e \in \{\rightarrow, \leftarrow\}) \implies \{(v_{z+1}, v, e) \mid v \in V'_z\}$ and $(v_{z+1}, v_G, \perp\!\!\!\perp) \implies \{(v_{z+1}, v, \perp\!\!\!\perp) \mid \forall v \in V'_z\}$ then by the base case we need at least one experiment to perform. □

Now that we have the bounds of our optimal set how do we go about discovering it? An intuitive approach may be what was formulated in Section 3.4.1.1: For each time step, choose the experiment with the highest amount of entropy [MWW17a, VJM00, MWW21]. The experiment with the highest entropy will most likely reduce the overall entropy of the graph and converge to the most certain outcome of a single graph at a higher rate than random. We discover entropy to be a matter of looking at degrees of freedom for each possible node and each possible outcome (DOF). Each node pair (v_1, v_2) initially contains three degrees of freedom, 1 for $v_1 \rightarrow v_2$, 1 for $v_2 \leftarrow v_1$ and the third for $v_1 \perp\!\!\!\perp v_2$ (no relation). Initially, the degrees-of-freedom heuristic will choose an experiment at random, perform it, then choose the next experiment based on minimizing entropy. We formalize this as:

```
$E^* \leftarrow$ Set of all possible edges
Choose $e_0 \sim$ Uniform$(E^*)$
$r \leftarrow$ Bernoulli$(p_{e_i})$
**if** $r = 1$ **then**
    $E^- \leftarrow \{\Delta \in E^* \mid e_i \in \Delta\}$
**else**
    $E^- \leftarrow \{\Delta \in E^* \mid e_i \notin \Delta\}$
**end if**
**while** $|E^*| > 0$ **do**
    Choose $e_i \sim$ Entropy$(E^*)$
    $r \leftarrow$ Bernoulli$(p_{e_i})$
    **if** $r = 1$ **then**
        $E^- \leftarrow \{\Delta \in E^* \mid e_i \in \Delta\}$
    **else**
        $E^- \leftarrow \{\Delta \in E^* \mid e_i \notin \Delta\}$
    **end if**
    $E^* \leftarrow E^* \setminus E^-$
**end while**
```

More details about the entropy calculation are given in Section 3.4.1. Although this approach seems better suited than choosing an experiment at random, there may be room for improvement.

Our previous work in Section 3.4.1.2 outlines another approach that involves quantifying expectation and making the choice that maximizes this metric (DOF+E). This expectation is a balance between the number of graphs eliminated by taking a particular decision, and the amount of information gained by a particular outcome. The particular outcome's gain is then multiplied by the probability of that outcome to obtain the expected value. The expectation algorithm is identical to the entropy-based approach, with the exception of using the expectation calculation as opposed to the entropy calculation.

Experiment 3.4.2 shows us that both of these approaches yield more information to the discovery of the complete causal graph G'_z quicker than random experiment selection. However, they are not optimal approaches. Instead, we suggest a third alternative to random selection based on a concept we introduce as *inference potential*. We take the approach of fragmenting the node set into a set of connected graphs by naively selecting a non-intervention experiment to be performed for each edge. As with DOF and DOF+E, this is not the most optimal approach, rather it provides intuition for the techniques we use later in a more natural setting while outperforming the entropy approaches. By performing a non-intervention experiment against each edge, we can imply a causal relation among all edges and separate graph fragments which are not connected. The causally related edges are initially constructed as undirected and will become more directed as more experiments are performed. The first phase partitions the graph into a set of independent graphs that are all connected. We seek to find an edge pair (v_1, v_2) for each connected graph that implies a connection among a set of nodes. For all nodes downstream of v_1 (D) and upstream of v_2 (U) if $v_1 \to v_2$, then this implies $d \in D \to u \in U$. Thus, we can make inferences about the causal nature of the graph without the need for experimentation. In combination with the phase 1 non-intervention determination, the edge inference should theoretically save the scientist from performing some experiments. By maximizing the number of experiments not performed, we can arrive at a true causal graph G'_z in a smaller number of total experiments. An algorithmic description for scoring an edge set is given by Algorithm 3.5.1.

Upon further inspection, this greedy approach can be further improved by noticing that we can improve upon our selection by uncovering more important inference nodes. To introduce this intuition we give a small case study:

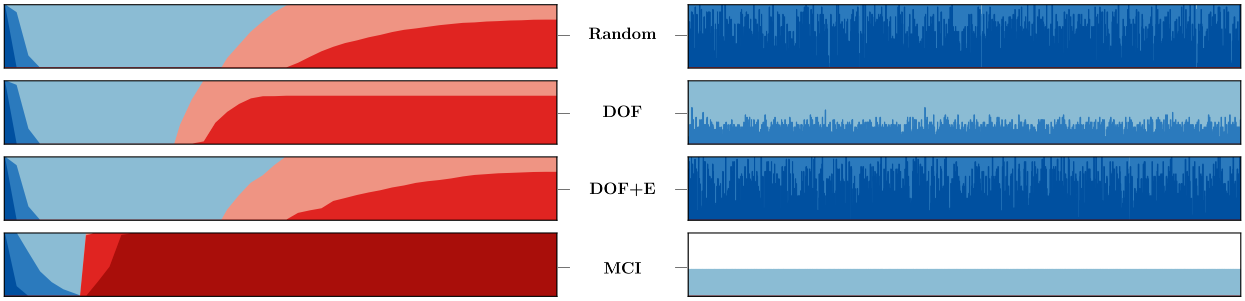


Figure 3.26: A comparison between baseline methods and Equation 3.117 showing the decreasing function of G'(a) and the total number of experiments needed to find G' in a 5,000 graph sample set over G^*_5.

3.5.1.6.1 Inference edge selection

We assume the causal graph we are working with contains 4 variables with the given true causal form as the first graph given in Figure 3.25. We run phase 1 of Algorithm 3.5.1 on the input set to get the dependent graph. The results of phase 1 leave us with the second graph given in Figure 3.25.

From here we diverge on two possible sets of choices. One set chooses to examine a side edge (top) which leads to the fourth graph. The other one chooses the middle graph which ends up relaying more information than the top choice. The reason being is that we can make inferences on the graph about upstream and downstream nodes. Since we know that no cycle can occur, and we know there is a causal relationship, we can assume the direction of the edge in the second graph of the bottom split whereas we cannot make any reducing assumptions in the top split.

The nature of the DAG also allows for another type of inference to be made related to cycles. If we can determine that a particular experiment would lead to a cycle, we can then eliminate that experiment. We propose another algorithm to prefer selecting experiments that would allow us to make eliminations based on this cycle constraint. This algorithm favors edge pairs contained in the highest number of cycles (given that some edge pairs are undirected). After calculating the number of cycles for each edge pair, we determine the causal relation for the edge pair contained in the highest number of cycles. We then repeat this process until the graph G'_z is discovered. To discover the number of cycles an edge pair (v_1,v_2) belongs to, we traverse all outgoing edges from v_2 using any graph traversal algorithm and count the number of times we arrive at v_1. However, in order to count the total number of cycles contained in (v_1, v_2) we must make a slight change to the graph traversal algorithm to allow for visiting the same node twice. This results in an $\mathcal{O}(n^2)$ running time. We store this number in the edge (v_1, v_2), choose the next edge, and repeat until all edges have been counted. The pseudo-code scoring algorithm is given in Algorithm 3.5.2. As given, Algorithm 3.5.2 is non-polynomial in execution time. To mitigate this, we select a threshold value for the max number of cycles to uncover and randomly choose among edges that meet the threshold—or the highest scoring edge, if no edges are above the threshold.

Both algorithms are combined, and the max scoring edge represents the next selected experiment. We refer to this combined algorithm as the max-cyclic-inference (MCI) algorithm. We seek a simple threshold (ψ) combination for the total score to combine both scoring methods. We assume to have a function $\mathcal{A}^+$ that takes as input an edge pair $e = (a, t)$, a graph G, and returns all the paths in G starting from a and ending with t and vice-versa. The ancillary function $\mathcal{A}$ will return each θ_e^c for each edge in $\mathcal{A}^+$ and c in the set of possible edge directions. With our assumed functions together with the transitive closure of G, G^+, we can express our combined scoring metric as follows:

$$e_i = (a_i, t_i) \tag{3.110}$$

$$G_i^- = \{d \in V \mid (d, a_i) \in G^+\} \tag{3.111}$$

$$G_i^- = \{u \in V \mid (t_i, u) \in G^+\} \tag{3.112}$$

$$f_i^d = \mathcal{A}(d, a_i, \theta, G) \tag{3.113}$$

$$f_i^u = \mathcal{A}(t_i, u, \theta, G) \tag{3.114}$$

$$M_i(e_i|G,\theta) = \sum_{d \in G_i^+} \sum_{u \in G_i^-} \prod_{\phi \in f_u^d} \phi\Big(\prod_{\omega \in f_i^u} \omega\Big) \overline{M_i}(e_i|G,\theta) = \frac{M_i(e_i|G,,\theta)}{\sum_{e \in E} M_i(e|G,\theta)} \tag{3.115}$$

$$M_c(e_i|G,\theta) = |\mathcal{A}(a_i, t_i, \theta, G)| \overline{M_c}(e_i|G,\theta) = \frac{M_c(e_i|G,,\theta)}{\sum_{e \in E} M_c(e|G,\theta)} \tag{3.116}$$

$$\mathcal{M}(e_i|G,\theta_{e_i}) = \psi \overline{M_i}(e_i|G,\theta) + (1-\psi)\overline{M_c}(e_i|G,\theta) \tag{3.117}$$

To show the utility of the MCI algorithm, we set up a simple experiment where we construct a true causal graph from the set of all causal graphs and run MCI against the three baseline algorithms

Algorithm 3.5.1 Max-Inference Scoring

Input: Connected graph $G = (V,E)$.

Output: A weighting for each Experiment e

```
procedure UPSTREAM($(v_1,v_2)$, $E$)
    enqueue $(v_2,\{v_1\})$ into $q$
    $u \leftarrow \varnothing$
    while $|q| > 0$ do
        $(v_c, s) \leftarrow$ dequeue from $q$
        if $\{v_x \mid (v_c, v_x \in E \wedge v_x \notin s\} = \varnothing$ then
            $u \leftarrow u \cup (s_0,s_1,\ldots,s_k,v_c)$
        else
            for $v_i \in \{v_x \mid (v_c, v_x \in E \wedge v_x \notin s\}$ do
                enqueue $(v_i,(s_0,s_1,\ldots,s_k,v_c))$ into $q$
            end for
        end if
    end while
    return $u$
end procedure
procedure DOWNSTREAM($(v_1,v_2)$, $E$)
    return equivalent to Upstream with the vertex positions swapped
end procedure
procedure PATHSCORE($d$, $u$, $E$)
    $\Sigma \leftarrow 0$
    for $i \leftarrow |d|$ to 0 do
        $p_d \leftarrow 1$
        for $s_i \in (d_i,d_{i-1},\ldots,d_0)$ do
            if $(s_i,s_{i-1},\rightarrow) \notin E$ then
                $p_d \leftarrow p_d \times 0.5$
            end if
        end for
        for $(u_0,u_1,\ldots,u_k) \in u$ do
            $p_u \leftarrow 1$
            for $j \leftarrow 0$ to $k$ do
                if $(u_j,s_{u+1},\rightarrow) \notin E$ then
                    $p_u \leftarrow p_u \times 0.5$
                end if
                $\Sigma \leftarrow \Sigma + p_d \times p_u$
            end for
        end for
    end for
    return $\Sigma$
end procedure
procedure SELECT($E$)
    $w \leftarrow \varnothing$
    for $(v_1,v_2) \in E$ do
        $u_r \leftarrow$ Upstream$((v_1,v_2),E)$
        $d_r \leftarrow$ Downstream$((v_1,v_2),E)$
        $u_l \leftarrow$ Upstream$((v_2,v_1),E)$
        $d_l \leftarrow$ Downstream$((v_2,v_1),E)$
        $w \leftarrow w \cup \{(v_1, v_2, \text{PathScore}(u_r, d_r, E) + \text{PathScore}(u_l, d_l, E))\}$
    end for
    return $w$
end procedure
```

Algorithm 3.5.2 Max-Cycle Scoring

Input: Connected graph $G = (V{,}E)$.
Output: A weighting for each Experiment e

```
procedure DFSCycle($(v_1{,}v_2)$, $r$, $E$, $S$)
    $c \leftarrow 0$
    for $(v_2, v_x) \in E$ do
        if $v_x = r$ then
            $c \leftarrow c + 1$
        else if $v_x \notin S$ then
            $c \leftarrow c +$ DFSCycle$((v_2{,}v_x), r, E, S \cup \{v_x\})$
        end if
    end for
    return $c$
end procedure
procedure Cycles($E$, $v_1$, $v_2$)
    if $(v_1{,}v_2, \rightarrow) \in E$ then
        return (DFSCycle$((v_1{,}v_2), v_1 E, \varnothing), 0)$
    else if $(v_1{,}v_2, \leftarrow) \in E$ then
        return $(0,$ DBFSCycle$((v_2{,}v_1), v_2, E, \varnothing))$
    else
        return (DFSCycle$((v_1{,}v_2), v_1, E, \varnothing)$, DFSCycle$((v_2{,}v_1), v_2, E, \varnothing))$
    end if
end procedure
procedure Select($E$)
    $w \leftarrow \{\}$
    for $(v_1{,}v_2) \in E$ do
        $w \leftarrow w \cup \{(v_1{,}v_2),$ Cycles$(E, v_1{,}v_2)\}$
    end for
    return $w$
end procedure
```

Algorithm 3.5.3 Experiment Selection

Input: Set of piecemeal causal variables V, weighting parameter ψ.
Output: Experiment e

```
procedure Select($V$,$\psi$)
    $w \leftarrow \{\}$
    $G^* \leftarrow \{\}$
    for $i \leftarrow 1$ to $|V|$ do
        for $j \leftarrow i + 1$ to $|V|$ do
            $r \leftarrow$ Result of experiment $(v_i, v_j, \not\leftrightarrow)$
            if $r = 1$ then
                $G_{i,j} \leftarrow \{G_k \in G^* \mid v_i \in V_k \vee v_j \in V_k\}$
                if $|G_{i,j}| > 0$ then
                    $V_{i,j} \leftarrow V_{i,j} \cup \{v_i, v_j\}$
                    $E_{i,j} \leftarrow E_{i,j} \cup \{(v_i, v_j, \not\leftrightarrow)\}$
                else
                    $G^* \leftarrow G^* \cup\ (\{v_i, v_j\}, \{v_i{,}v_j, \not\leftrightarrow\})$
                end if
            end if
        end for
    end for
    for $G_k \in G^*$ do
        for $(v_1{,}v_2) \in E_k$ do
            $w^i \leftarrow w_i \in$ Max-Inf$(E, v_1{,}v_2)\}$
            $w^c \leftarrow w_i \in$ Max-Cycle$(E, v_1{,}v_2)\}$
            $w \leftarrow w \cup \{(v_1{,}v_2), \psi w^i + (1 - \psi) w^c\}$
        end for
    end for
    return $max\{w_m \in w\}$
end procedure
```

(Random, DOF, DOF+E) and count the number of experiments conducted in order to discover G'. We then repeat this experiment for all $g \subset G_5^*$.

3.5.1.6.2 Experimental Setup

We choose an entity size of five nodes and randomly select 5,000 graphs from the total set of graphs in G_5^*. The total set comprises 29,281 graphs as the number of graphs follows from the following equation:

$$|G_z^*| = \sum_{m=1}^{z} (-1)^{m-1} \binom{z}{m} 2^{m(z-m)} |G_{z-m}^*| \tag{3.118}$$

For each $G' \in \{g \subset G_5^*\}$ we run the random algorithm (Random) against the degrees of freedom (DOF), expectation (DOF+E), and max-cyclic-inference (MCI) algorithms. For each algorithm, we count the number of experiments used to determine the true casual graph G'.

3.5.1.6.3 Experimental Results

The results of the experiment are visualized in Figure 3.26. Figure 3.26 (left) demonstrates the advantage of the MCI approach by comparing the average number of graphs set at time x against the baseline methods. To emphasize a small subset of possible graphs, we invert and scale the count of graphs less than 10. As we can see from Figure 3.26 (left), MCI initially takes longer to reduce possible graphs size; but after an initial lagging, it drastically outperforms all the other algorithms. This outperformance can be seen in the inverted portion of the horizon plot, which shows that the number of experiments needed to obtain a small set of possible graphs is significantly lower than that of the other methods. Moreover, when it reaches the inversion threshold, it reduces to a smaller possible set size than the other methods.

As shown in Figure 3.26 (right), the other result we compare is the total number of experiments performed for each causal graph in G'. An interesting finding is that the expectation algorithm is marginally better than Random, and less better than the degree-of-freedom (DOF) approach. This finding differs slightly from the results in Experiment 3.4.2. We hypothesize this result is due to the increase in the graph dimension size of G_5^*, suggesting that DOF may be a more informative algorithm than the expectation-based approach. Out of all the algorithms, MCI once again outperforms the baseline methods by a large margin. This result implies our algorithm may be useful to help guide scientists in selecting subsequent experiments, given a partial causal graph.

3.5.1.7 The Global Causal Graph

So far, our discussion on experiment selection has made some basic assumptions about the casual world: e.g., that a causal relation exists distinctly in one of three classes, and that there are no causal cycles in the true graph. In the actual biological setting, things are not so simple—we may indeed observe conflicting results across two iterations of the same experiment, under ostensibly identical conditions. Conflicting evidence may lead to uncertainty about the edge relationship, including whether cycles govern the system's behavior. We propose a remedy for these scenarios is given by leveraging the CPCEI.

We can take each experiment as an observation from a hidden variable, which we update using Bayesian analysis (CPCEI). This method also allows us to formulate a measure of certainty, which is the probability of a given edge belonging to a particular class. Initially, each probability is given an uninformative prior and is updated as we record observations. Under this model, each edge becomes a probability distribution, and we permit cycles. This concept of a probability for a causal outcome with cycles leads to the interesting extension of *piecemeal causal decay*.

Definition: piecemeal causal decay. Piecemeal causal decay (PCD) is the likelihood that two biological entities have a causal effect on each other given the set of known paths with the same causal effect, originating from v_1 and ending at v_2. For example, if at a moment in time, the entirety of the known world exists such that there is a causal relationship of $A \rightarrow B \rightarrow C \rightarrow D$, then we can imply a stronger casual likelihood of $A \rightarrow C$ than $A \rightarrow D$.

Given our assumption of a biological decay associated with the path length of a causal chain, we model this as a joint probability distribution. In the example above, if the $p(A \rightarrow B)$ is 0.5 and the $p(B \rightarrow C)$ is 0.75 then we take the $p(A \rightarrow C)$ as $0.5 \times 0.75 = 0.375$. Under this model, the likelihood of an association is a naturally decreasing function that we assume mimics what is found in the scientific setting. This model also adds an interesting property to that of measuring the strength of information about a particular causal pair: that any observed causal experiment's result is an observation of not only the two edge pairs being studied but for every causal chain that contains v_1 and v_2. We can thus update not only the score between v_1 and v_2 but every downstream biological entity. Biological and piecemeal casual decay leads us to another interesting concept: *potential piecemeal causation.*

Definition: potential piecemeal causation. For each element along a causal chain that is not previously studied, we can apply the concept of PCD to imply a causal relationship for a pair of variables. This implied causal relationship represents the potential for causation, an assertion we have tangential knowledge for, but needs to be verified experimentally.

Much like the score we arrive at for piecemeal causation between two studied entities, we can calculate the potential piecemeal causation (PPC) as a likelihood defined by the PCD of each element. This calculation then becomes product of CPCEI values along each path:

$$\mathcal{P}_{i,j} = \{v_i, v_{i+1}, \ldots, v_j\} \tag{3.119}$$

$$\mathcal{L}((v_i, v_j, c) \mid E) = \prod_{v \in \mathcal{P}_{i,j}} \frac{\alpha_v + n_x^{v,c}}{\sum_i^J \alpha_i + n_x^{v,c}} \tag{3.120}$$

Even though potential causation is supported only by tangential evidence, we take direct evidence to be much stronger. Because we value direct evidence much stronger, we must determine a correct mechanism for assigning a stronger weight to such evidence. An interesting observation of potential causation's calculation leads to the natural weighting of direct versus tangential evidence. The multiplication of probabilities along each edge pair will decrease significantly, allowing direct evidence to receive a more considerable weight. Again, we think that this mirrors the scientific setting.

To factor causal potential into the score, we take the sum total of potential causation plus the score given by the CPCEI as the value of a given casual strength and likelihood, but this becomes a recurrent relationship. As a given score changes, so too does downstream and upstream scores. One approach to take is to use an iterative algorithm to calculate the score. At time x_0, we set the scores to the direct causal score given in the CPCEI. Then for each edge pair and time x, we calculate the given score by multiplying the scores derived at time $x - 1$ on the edges for each path that passes through (v_1, v_2). The next time step continues the same calculation until the values converge.[4]

[4] For simplicity, in our experimentation we limit the the number of timesteps to 2.

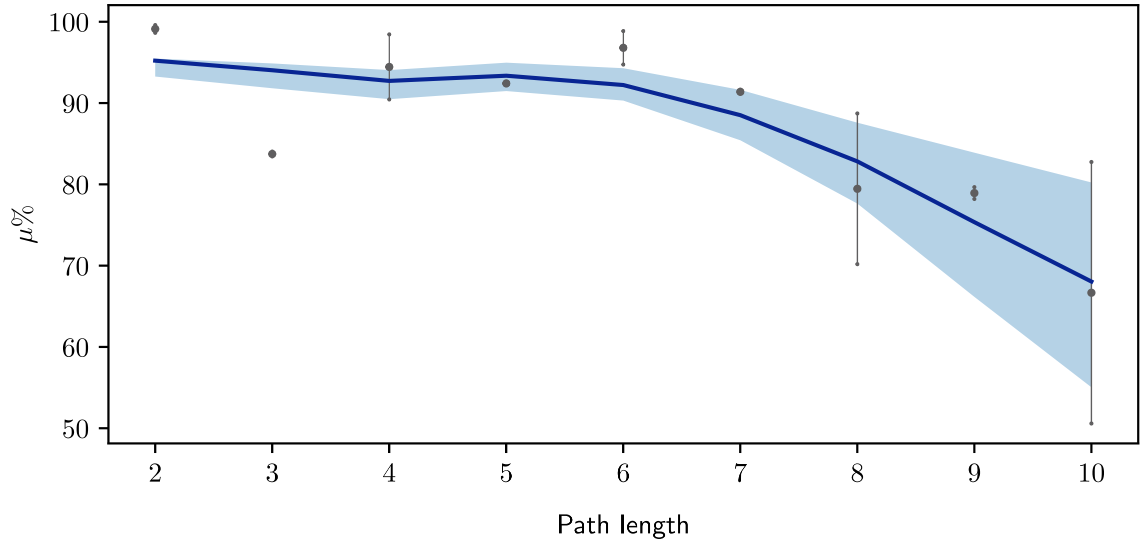


Figure 3.27: Loess smoothed regression and 95% confidence intervals describing the likelihood of a causal observation given an observed path length.

To help illustrate the usefulness of our probabilistic scoring for experiment selection, we give a case study using a small study of published research papers over a period of 17 years.

3.5.1.7.1 Case Study

We take a small set of papers consisting of 5 research papers published from 1997 to 2011. From these papers, we extract out a sequence of experiments that connect a later research paper. The papers are given in the following table: The sequence of experiments yielded, in order, the edges

Agent	Target	Year
NF1	Spatial learning (SL)	1997 [SFM97]
SL	PERK	2005 [KEM05]
PERK	LTP	2006 [COG06]
LTP	Arc	2011 [KF11]
Arc	Spine density (SD)	2011 [KF11]

NF1 → SL [SFM97] → PERK [KEM05] → LTP [COG06] → Arc [KF11] → SD [KF11]. We can use our calculation to place a more likely causal association between NF1 and LTP than between NF1 and SD, which indeed was suggested by empirical evidence in 2013 for both assertions [SHM13].

This example highlights a specific instance where our model correctly predicts the outcomes of experiments, but it is worth investigating whether this is generally likely. To investigate this, we set up the following experiment.

3.5.1.7.2 Experimental Setup

From our entire aggregation of piecemeal causal assertions, we find every path of length $n \leq 10$ and record the result of the experimental assertion, with the start of the path being the agent and the end of the path being the target. We then record the average number of causal observations.

3.5.1.7.3 Experimental Results

The average number of causal observations (μ) are plotted alongside the path length in Figure 3.27. We also record the 95% confidence interval for each μ. To show the general trend, we use a loess-based regression smoothing. As we can see from the figure, there is a general downward trend. As we increase the path length, we tend to see less causal relationships. Although these results are somewhat complicated by the assumption that a no relationship result is less likely to be reported, they nevertheless support our concepts of PCD and PPC.

So given the concept of a causal score, casual decay, and potential causation coupled with a partially constructed view of the causal world, an interesting area to investigate is that of experiment selection. Given a large set of possible experiments to take, what should be the next experiment chosen? This question is indeed important as experiments in the biological setting can be arduous, expensive, and time-consuming. One simple approach would be techniques that mimic entropy and entropy expectation, as discussed in Section 3.5.1.6. However, we may benefit in the same manner by choosing an experiment taken from scores provided by a combination of the CPCEI and PPC (CPCEI-P). We take this combination to be a simple addition of values returned by each metric:

$$s((v_i, v_j) \mid c, E, \theta^c_{i,j}) = \mathbb{E}[\theta^c_{i,j} \mid n^{i,j,c}_x, \alpha_{i,j}] + \mathcal{L}((v_i, v_j, c) \mid E) \tag{3.121}$$

We seek to develop an analogous approach to Equation 3.117. But this is a bit challenging due to the relaxed constraints of distribution edge scoring and cycles. For the purpose of discovering the next experiment to take, we factor into the model a set of independent graphs in our global causal graph. We can declare a non-edge to be an edge where the evidence is not existing. With each sub graph we can isolate each individual class and calculate the amount of score to be gained as in Equation 3.117. We then take total value to be that of the calculated score returned by Algorithm 3.5.1.6.1. We can formulate an adaption of the MCI that seeks out the maximum score gained, where the score is calculated using Equation 3.121. This equation is given as (e_i is defined in Equation 3.110):

$$\theta_{e_i}^{c,1} = \mathbb{E}[\theta_{e_i}^{c} \mid n_x^{e_i,c} + 1, \alpha_{e_i}] \tag{3.122}$$

$$g_c(e_i|G, \theta_{e_i}^{c}) = |s(e_i|c, E, \theta_{e_i}^{c,1}) - s(e_i|c, E, \theta_{e_i}^{c})| \tag{3.123}$$

We can further our existing approach by considering the nature of our causal world—that is, a set of massively connected graphs. To fully discover the true nature of these graphs requires a tremendous amount of experiments to be conducted. And due to the constraints of conducting experiments, it is not feasible for any team of scientists, human or robot, to discover the true global graph in totality. We must therefore consider the effect an experimental result makes on the set of causal graphs. An interesting observation can be made that improving the causal knowledge of an existing edge pair can affect connecting edges to either biological element (node) score calculation. In other words, if the result of an experiment e_1 yields a particular result then another connected experiment e_2 may become more advantageous to take than before conducting e_1. This leads to a realization that two edge pairs (v_1, v_2) and (v_3, v_4) with the same initial score may not be equal in terms of being a preferable choice. If (v_1, v_2) leads to a causal graph where another highly valued edge pair (v_5, v_6) increases in value to a value higher than any other edge pair after choosing (v_3, v_4), we would prefer to choose (v_1, v_2)—given only two choices to make. We submit that there is a potential for a cascading effect—in more detail: choosing one experiment may result in another experiment being more attractive than before the first experiment was selected.

Though it may be an exponential solution to solve optimally, we provide a heuristic algorithm for choosing the next best edge pair, which can be interpreted as the next best experiment to perform in the context of piecemeal causal discovery. Given all calculated scores for all edge pairs in a causal graph for class c, we place the edge weights into a modified edge-based PageRank. We then run the PageRank algorithm until convergence and record all the values. We repeat this process for all classes and record an edge score for each edge as the sum of all PageRank values for each class normalized by the number of classes. Our modified PageRank algorithm is calculated as:

$$R(v_j) = \frac{1-d}{|V|} + d \ \cdot \sum_{v_i \in I(v_j)} \frac{\mathcal{G}(v_i,v_j)}{\sum\limits_{v_o \in O(v_i)} \mathcal{G}(v_i,v_o)} \cdot R(v_i) \tag{3.124}$$

Where d is a damping factor, $I(v_j)$ the set of incoming nodes into v_j, $O(v_i)$ the set of outgoing nodes from v_i, and $\mathcal{G}(v_i, v_j) = g_c(e_b|G, \theta^c_{e_b})$ from Equation 3.123 with e_b being the edge between (v_i, v_j). After the final scores for each edge are calculated, we take the highest-scoring edge; if there are multiple high-scoring edges, we randomly choose among the highest-scoring edges. After conducting the experiment, we update the scores of the graph and rerun Equation 3.124. The full algorithm is shown in Algorithm 3.5.4.

Given our approaches to determine the next best experiment, we demonstrate the choices of the different approaches on our existing snapshot of the casual world. We use the causal graph described in Section 3.5.1.4, and the algorithms in Section 3.5.1.6 and Algorithm 3.5.4 to determine and report the experiments we recommend biologists pursue.

3.5.1.7.4 Experimental Setup

We build our causal graph from the entire ResearchMaps database coupled with the causal datasets described in Table 3.2 and the discovered relationships from Experiment 3.5.1.2.1. The edge weights are restricted to only the evidence gathered, and initially do not factor in potential causation. With the initial state of the causal graph representing our best guess of the causal world, we score each edge with the algorithms described in Section 3.5.1.6 and Algorithm 3.5.4 to determine the next best experiment to conduct.

Algorithm 3.5.4 Experiment Selection

3.5.4 **Input:** Causal graph G.
Output: Next best experiment E

```
procedure SCORE($G$,($e_1$,$e_2$),$p$)
    $s^{u}_{\rightarrow} \leftarrow$ DirScore($G$, ($e_1$,$e_2$), $\rightarrow$, $s_{(e_1,e_2)}$)
    $s^{d}_{\rightarrow} \leftarrow$ RevScore($G$, ($e_1$,$e_2$), $\rightarrow$, $s_{(e_1,e_2)}$)
    $s^{u}_{\leftarrow} \leftarrow$ DirScore($G$, ($e_1$,$e_2$), $\leftarrow$, $s_{(e_1,e_2)}$)
    $s^{d}_{\leftarrow} \leftarrow$ RevScore($G$, ($e_1$,$e_2$), $\leftarrow$, $s_{(e_1,e_2)}$)
    $p^* \leftarrow \overrightarrow{s}$ according to Equation 3.121
    $s^{u*}_{\rightarrow} \leftarrow$ DirScore($G$, ($e_1$,$e_2$), $\rightarrow$, $p^*$)
    $s^{d*}_{\rightarrow} \leftarrow$ RevScore($G$, ($e_1$,$e_2$), $\rightarrow$, $p^*$)
    $s^{u*}_{\leftarrow} \leftarrow$ DirScore($G$, ($e_1$,$e_2$), $\leftarrow$, $p^*$)
    $s^{d*}_{\leftarrow} \leftarrow$ RevScore($G$, ($e_1$,$e_2$), $\leftarrow$, $p^*$)
    if $p = \rightarrow$ then
        return $p \times |(s^{u}_{\rightarrow} + s^{d}_{\rightarrow}) - (s^{u*}_{\rightarrow} - s^{d*}_{\rightarrow})|$
    end if
    if $p = \leftarrow$ then
        return $p \times |(s^{u}_{\leftarrow} + s^{d}_{\leftarrow}) - (s^{u*}_{\leftarrow} - s^{d*}_{\leftarrow})|$
    end if
    $\delta_1 \leftarrow p \times |(s^{u}_{\rightarrow} + s^{d}_{\rightarrow}) - (s^{u*}_{\rightarrow} - s^{d*}_{\rightarrow})|$
    $\delta_2 \leftarrow p \times |(s^{u}_{\leftarrow} + s^{d}_{\leftarrow}) - (s^{u*}_{\leftarrow} - s^{d*}_{\leftarrow})|$
    return $p \times (\delta_1 + \delta_2)$
end procedure
procedure DIRECTIONALITY($\overrightarrow{s}$)
    $p_m \leftarrow$ max($\overrightarrow{s}$)
    if $p_{\nleftrightarrow} = p_m$ then
        return $\nleftrightarrow$
    end if
    if $p_{\rightarrow} = p_m$ then
        return $\rightarrow$
    end if
    return $\leftarrow$
end procedure
procedure NEXT($G$)
    for $(e_1, e_2) \in G$ do
        Calculate $\overrightarrow{s}$ according to Equation 3.121
        Directionality $\leftarrow$ Directionality($\overrightarrow{s}$)
    end for
    $m_s \leftarrow 0$
    $m_e \leftarrow \varnothing$
    for $(e_1, e_2) \in G$ do
        for $p \in \theta_w$ do
            $s \leftarrow$ Score($G$, $(e_1, e_2)$, $p$)
            if $s > m_s$ then
                $m_s \leftarrow s$
                $max_e \leftarrow (e_1, e_2, p)$
            end if
        end for
    end for
    return $m_e$
end procedure
```

3.5.1.7.5 Experimental Results

We display the top seven experiments to conduct in Table 3.9 for Equation 3.124 and Algorithm 3.5.4. We can see a slight difference between the two scoring metrics and expect Equation 3.124 to be the most impactful. However, we hope that the biological community can confirm this presumption.

Algorithm 3.5.5 Edge Scores

Input: Causal graph G.
Output: Next best experiment E

```
procedure DIRSCORE($G$,($e_1$,$e_2$),$d$,$\delta$)
    mark $e_2$ as visited
    sum $\leftarrow -\delta$
    enqueue ($e_1, \varnothing, \delta$ into $q$
    while $|q| > 0$ do
        $(n, \_, s) \leftarrow$ dequeue from $q$
        if $s < \lambda$ then
            continue
        end if
        $sum \leftarrow sum + s$
        mark $n$ as visited
        for unvisited $e_i \in \{(n, e_i) \in G \cup \{n, e_i\} \in G\} \wedge (n,e_i) = d$ do
            enqueue $(e_i, n, s \times s_{(n,e_i)})$ into $q$
        end for
    end while
    return sum
end procedure
procedure REVSCORE($G$,($e_1$,$e_2$),$d$,$\delta$)
    mark $e_2$ as visited
    sum $\leftarrow -\delta$
    enqueue ($e_1, \varnothing, \delta$ into $q$
    while $|q| > 0$ do
        $(n, \_, s) \leftarrow$ dequeue from $q$
        if $s < \lambda$ then
            continue
        end if
        $sum \leftarrow sum + s$
        mark $n$ as visited
        for unvisited $e_i \in \{(e_i, n) \in G \cup \{e_i, n\} \in G\} \wedge (e_i, n) = d$ do
            enqueue $(e_i, n, s \times s_{(e_i,n)})$ into $q$
        end for
    end while
    return sum
end procedure
```

The experiment above highlights the best experiment to take to increase the overall knowledge of our casual world, but we can use similar techniques for other types of knowledge discovery. We demonstrate this in the discovery of likely causal associations.

3.5.1.7.6 Experimental Setup

Using the same approach as described in Experiment 3.5.1.7.4, after the scoring of the graph is complete using Algorithm 3.5.4, we modify our search to only elements with no direct observations. We perform the search for each of the three types of causal classes.

Equation 3.124			Algorithm 3.5.4		
Agent	**Target**	**Score**	**Agent**	**Target**	**Score**
sharp-wave ripples	oAβ	8.86e-4	glutamate release	mGluR7	3.5e-3
myosin motor proteins	actin cytoskeleton	4.91e-4	contextual conditioning	HIV Tat	3.5e-3
AMPAR	memory retrieval	4.75e-4	CRE	NMDA	3.35e-3
pSer845	memory retrieval	4.64e-4	CDK5	NMDA	3.12e-3
PGE2 generation	TNFR1 activation	4.56e-4	glutamate release	NF1	2.9e-3
TTR	spatial memory	4.26e-4	hippocampal memory	KIF17	2.8e-3
presynaptic GABA release	mGluR7	4.25e-4	pERK	pSynapsin I	2.62e-3
TNF-α	PGE2 generation	4e-4	TORC1	pCREB	2.6e-3
proBDNF	microglia	3.93e-4	BDNF	GP120	2.32e-3

Table 3.9: Experiment recommendations and scores for the most informative experiment to select for Equation 3.124 and Algorithm 3.5.4.

3.5.1.7.7 Experimental Results

We show the results of our search in Table 3.10. We believe the top-scoring biological elements, shown in Table 3.10, to most likely yield a causal relationship if tested directly. Discovering these types of associations may be more meaningful to biologists as it has the potential for the discovery of new treatments and therapies.

Another interesting discovery we can make is that of the likelihood of direct causation. That is, given an observation of causation between two elements (v_1, v_2), how likely is the effect being shown due to a direct association between the two elements versus confounding variables? Although this is a difficult question to answer, we posit the evidence of such a claim can be demonstrated by examining upstream and downstream elements connected to $e_i = (a_i,t_i)$. For example if we are given that $v_1 \to v_2$ and also $v_1 \to v_3$, if $v_3 \not\to v_2$ then this is evidence of a direct connection. Of course we do not have singular assignments of any causal class so we take the probability of causation as the assignment. If we take the aggregate over all elements we can rank the edges by the highest score and take these to be most likely of association. Conversely the lowest ranking score

Most likely to observe a relation	Most likely directly connected	Least likely directly connected
CXCR4 → spatial learning	IFN-γ → IDO1	pCRTC1 → contextual conditioning
TORC1 → pCREB	FLNA → dendritic complexity	TSC2 → MAPK
BDNF mRNA → pCREB	GRIN2B → hippocampal memory	intracellular calcium → CREB-dependent transcription
PKA → CTA	BDNF → pTrkB	CaMKII → spatial learning
NF1 → glutamate release	CCR5 → MAPK-CREB signaling	pCRTC1 → cFos
NF1 → pERK	insomnia → depression	TSC2 → UCP2
c-Fos → mTOR	pERK2 → pCREB	MEK → pERK
NDDs → cortical patches	CREB → 14.3.3 eta	FMRP → protein synthesis
TNF-α release → CXCR4	RAS → RAF kinase	TSC2 → mTOR

Table 3.10: Agent and target pairs that we suggest are most likely to reveal a causal relation (left), most likely to be directly connected (middle) and most likely for the observed causal effect to be due to confounding variables (right).

elements are either elements without a lot of evidence, indicating the need for more experimentation, or have contrary evidence. We quantify this in the following equation, given G^+ as the transitive closure of G:

$$U_i^c = \{\theta_j^c \mid (a_i, j) \in G^+ \land (j, t_i) \in E\} \tag{3.125}$$

$$D_i^c = \{\theta_q^c \mid (q, a_i) \in G^+ \land (q, t_i) \in E\} \tag{3.126}$$

$$M_s(e_i|G, \theta_i^c) = \theta_i^c - \Big(\sum_{h \in U_i^c} h + \sum_{l \in D_i^c} l\Big) \tag{3.127}$$

From the highest-scoring contrarian evidence, we posit these elements as evidence of no direct association—that is, elements that may lead to an observed effect, but due more to confounding variables than to a direct connection among themselves.

3.5.1.7.8 Experimental Results

For each edge in our global causal graph with a direct observation, we score the edge given by Equation 3.127. We score each causal class and report the results in Table 3.10. This experiment relays another interesting area for scientists to look at, as it may be useful to target these specific nodes (target) when investigating an effect of an agent. Likewise, we examine the lowest scoring edge pairs as another avenue for investigation. If a scientist knows that two elements, while on the surface providing evidence of causation, are due to a confounding variable, the scientist can search for that confounding variable, disconnect it from the casual chain, and test the causal relation between the original two elements and confirm or deny the supposition. This type of discovery is equally valuable since it can improve upon existing drugs and therapies aimed at the agent, which may not be as effective as targeting another agent.

3.5.2 Discussion

Here we present a pipeline that can serve as the "mind" of a robot scientist. We demonstrate the utility of this pipeline when applied to the PubMed Central corpus, though the techniques should remain useful in other datasets and domains. Since causality is prevalent in most areas of science, it is conceivable that our methods could also apply to these areas.

Although we demonstrate how the causal pipeline can improve scientific experimentation, our method is far from complete, with many areas that can be improved. One such area is the extraction of causal statements: our method aligns candidate sentences with a smaller labeled data set. An overview of improvements on our alignment method is discussed in Section 3.2.3. Another area of future research is the improvement of our probabilistic graph-scoring calculation. Our existing method assumes that any reported association is of equal strength. However, this assumption is not valid in the scientific setting, where some relationships may be demonstrated using a more statistically robust study. In this regard, it may be interesting to incorporate reported strengths of assertion, such as p-values.

Future work could also improve upon our experiment-selection algorithms. Even in our given setting, we make no claim of optimality—leaving open the possibility of improvements, which are likely to have a significant impact in the biological domain. There may also be other interesting areas to discover in regards to our causal graph. One area we did not address here was ranking the importance of a node in a causal network. It could be possible to apply a PageRank type algorithm to a causal subgraph to determine the most important node—and then this important node could be the focus of experimentation which could yield improved treatments.

This work explores a knowledge-synthesis, causal-discovery, and experiment-selection pipeline, which we propose as the "mind" for a robot scientist. We demonstrate the utility of the pipeline by applying it to the PubMed dataset. Our pipeline consists of a combination of established techniques, together with some novel improvements. Starting with the extraction of causal information, we feed this data into a technique that represents the information graphically. With a causal graph constructed, we can use graph-based knowledge discovery together with a novel scoring approach to yield findings that may be tremendously useful and impactful to the biological scientist. Our main contribution is demonstrating experimentally the benefit of adopting such an approach. When provided real-life data, the pipeline can produce interesting findings. We share the knowledge discovery demonstrated by our pipeline with the intention of providing immediate benefit to the biological research community.

CHAPTER 4

Topic Interpretability and Labeling

Existing approaches to Bayesian topic modeling are often based off Latent Dirichlet allocation (LDA) [BNJ01] and involve analyzing a given corpus to produce a distribution over words for each latent topic and a distribution over latent topics for each document. The distributions representing topics are often useful and generally representative of a linguistic topic. Unfortunately, assigning labels to these topics is often left to manual interpretation.

Identifying topic labels is useful in summarizing a set of words comprising a topic. For example, words of a topic, such as the words pencil, laptop, ruler, eraser, and book can be mapped to the label "School Supplies." Adding descriptive semantics to each topic can help people, especially those without domain knowledge, to understand topics obtained by topic modeling.

A motivating application of accurate topic labeling is to develop summarization systems for primary care physicians, who are faced with the challenges of being inundated with too much data for a patient and too little time to comprehend it all [MRD06]. The labels can be used to more appropriately and quickly give an overview, or a summary, of patient's medical history, leading to better outcomes for the patient. This added information can bring significant value to the field of clinical informatics which already utilizes topic modeling without labeling [AEB10, BLF11, SOA16].

Existing approaches in labeling topics usually do their fitting of labels to topics after completion of the unsupervised topic modeling process. A topic produced by this approach may not always match well with any semantic concepts and would therefore be difficult to categorize with a single label. These problems are best illustrated via a simple case study.

4.0.0.1 Case Study

Suppose a corpus of a news source that consists of two articles is given by documents $d1$ and $d2$, each with three words:

$\mathbf{d1}$ - pencil, pencil, umpire

$\mathbf{d2}$ - ruler, ruler, baseball

LDA (with the traditionally used collapsed Gibbs sampler, standard hyperparameters and the number of topics (K) set as two) would output different results for different runs due to the inherent stochastic nature. It is very possible to obtain the following result of topic assignments:

$\mathbf{d1}$ - pencil^1, pencil^1, umpire^2

$\mathbf{d2}$ - ruler^2, ruler^2, baseball^1

But these assignments to topics differ from the ideal solution that involves knowing the context of the topics in which these words come from. If the topic modeling was to incorporate prior knowledge about the topics “School Supplies” and “Baseball”, then a topic modeling process will more likely generate the ideal topic assignments of:

$\mathbf{d1}$ - pencil^2, pencil^2, umpire^1

$\mathbf{d2}$ - ruler^2, ruler^2, baseball^1

and assign a label of “School Supplies” to topic 1 and “Baseball” to topic 2. Furthermore, it is advantageous to incorporate this prior knowledge during the topic modeling process. Consider the following table displaying four different mapping techniques of the first result using the Wikipedia articles of “School Supplies” and “Baseball” as the prior knowledge ([same] means that in different runs the labels for Topic 1 and Topic 2 changed, however they were always equal to each other):
Applying this labeling post topic modeling can lead to problems dealing with the topic themselves. This is not so much a problem of the mapping techniques but of the topics used as input. By separating the topics during inference this problem of combining different semantic topics can be avoided. Equally important to the task of accurately labeling topics is that the most assigned words

Technique	Topic 1	Topic 2
JS Divergence	Baseball	Baseball
TF-IDF/CS	[same]	[same]
Counting	Baseball	Baseball
PMI	[same]	[same]

in the labeled topic are semantically related to the label. The semantic relatedness of the words most represented in the topic is referred to as topic interpretability. An approach that improves topic labeling and topic interpretability is an ideal of our topic model—a desiderata that is much more important than predictability when the topic model distributions are to be consumed by humans.

To label topics and increase interpretability, one may take a weakly-supervised approach that incorporates prior knowledge into the topic modeling process to improve the quality of token assignments and more effectively label topics. However, existing supervised approaches [SSC11, HRS13, JIU12] are either too lenient or too strict. For example, in the Concept-topic model (CTM) [SSC11], a multinomial distribution is placed over known concepts with associated word sets. This pioneering approach does integrate prior knowledge, but does not take into account word distributions. For example if a document is generated about the topic "School Supplies" it is much more probable to see the word "pencil" than the word "compass," even though both words may be associated with the topic "School Supplies." This technique also requires some supervision which requires manually inputting preexisting concepts and their bags of words.

Another approach, given by Hansen et al. as the explicit Dirichlet allocation (EDA) [HRS13], incorporates a preexisting distribution based off Wikipedia articles but does not allow for variance from the Wikipedia article distribution (formed by the word-count histogram). This approach fulfills the goal of incorporating prior knowledge and their distributions, but assumes a subset of topics in the generated corpus (corpus for which we are to perform topic modeling) strictly follow the Wikipedia word distributions.

To allow for more flexibility while leveraging the influence from weakly-supervised word distributions, we propose the Source-LDA model which is a balance between the CTM and EDA. The goal is to allow for simultaneous discovery of both known and unknown topics. Given a

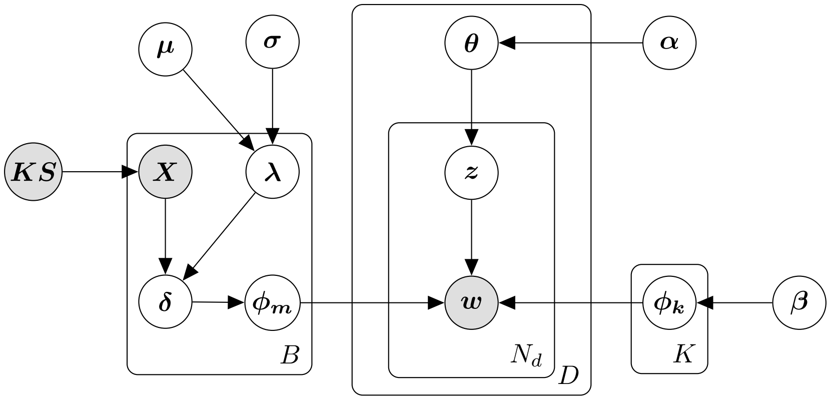


Figure 4.1: Plate notation Source-LDA.

collection of known topics and their word distributions, Source-LDA is able to identify the subset of these topics that appear in a given corpus. It allows some variance in word distributions to the extent that it optimizes the topic model.

4.1 Source-LDA

Source-LDA is an extension of the LDA generative model. In Source-LDA, after a known set of topics are determined, an initial word-to-topic distribution is generated from corresponding Wikipedia articles. The desiderata is to enhance existing LDA topic modeling by integrating prior knowledge into the topic modeling process. The relevant terms and concepts used in the following discussion are defined below.

Definition 1 (Knowledge source)**.** *A knowledge source is a collection of documents that are focused on describing a set of concepts. For example the knowledge source used in our experiments are Wikipedia articles that describe the categories we select from the Reuters dataset.*

Definition 2 (Source Distribution)**.** *The source distribution is a discrete probability distribution over the words of a document describing a topic. The probability mass function is given by*

$$\forall w_i \in W,\ \ f(w_i) = \frac{n_{w_i}}{\sum_j^G n_{w_j}}$$

where W is the set of all words in the document, $G = |W|$, and n_{w_i} is the number of times word w_i appears in the document.

Definition 3 (Source Hyperparameters)**.** *For a given document in a knowledge source, the knowledge source hyperparameters are defined by the vector* $(X_1, X_2, \ldots, X_V)$ *where* $X_i = n_{w_i} + \epsilon$ *and* ϵ *is a very small positive number that allows for non-zero probability draws from the Dirichlet distribution.* V *is the size of the vocabulary of the corpus for which we are topic modeling, and* n_{w_i} *is the number of times the word* w_i *from the corpus vocabulary appears in the knowledge source document.*

We detail three approaches to capture the intent of Source-LDA. The first approach is a simple enhancement to the LDA model that allows for the influencing of topic distributions, but suffers from needing more user intervention. The second approach allows for the mixing of unknown topics, and the third approach combines the previous two approaches. It moves toward a complete solution to topic modeling based off prior knowledge sources.

4.1.1 Bijective Mapping

In the simplest approach, the Source-LDA model assumes that there exists a 1-to-1 mapping between a known set of topics and the topics used to generate a corpus. The generative model then assumes that instead of selecting topic-to-word distributions from a sampling from the Dirichlet distribution, a set of K distributions are given as input and sampled from after each topic assignment is sampled for a given token position. The generative process for a corpus adapted from the traditional LDA generative model during the construction of the ϕ distributions is as follows (for brevity only the relevant parts of the existing LDA algorithm given in Section 2.2.1.2 are shown):

```
1.  For each of the $K$ topics $\phi_k$:
2.      $\delta_k \leftarrow (X_{k,1}, X_{k,2}, \ldots, X_{k,V})$
3.      Choose $\phi_k \sim \text{Dir}(\delta_k)$
```

Where $(X_{k,1}, X_{k,2}, \ldots, X_{k,V})$ represents the knowledge source hyperparameters for the k_{th} knowledge source document. The generative model only differs from the traditional LDA model in how each ϕ is built. Therefore the derivation for inference requires only a simple modification. To approximate the distributions for θ and ϕ, a collapsed Gibbs sampler can approximate the z assignments as follows:

$$P(w_i|z_i{=}j, z_{\text{-}i}, w_i) \propto P(w_i|z_i{=}j, z_{\text{-}i}, w_{\text{-}i}) P(z_i{=}j|z_{\text{-}i}) \tag{4.1}$$

Following the Bayesian model (Figure 4.1), the following equations can be easily be generated

$$P(w_i|z_i{=}j,z_{\text{-}i},w_{\text{-}i}){=}\int P(w_i|z_i{=}j,\phi_j)P(\phi_j|z_{\text{-}i},w_{\text{-}i})d\phi_j \tag{4.2}$$

with

$$P(\phi_j|z_{\text{-}i},w_{\text{-}i}) \propto P(w_{\text{-}i}|\phi_j,z_{\text{-}i})P(\phi_j) \tag{4.3}$$

$$P(\phi_j|z_{\text{-}i},w_{\text{-}i}) = Dir(\delta_{i,j}+n_{w_{\text{-}i,j}}) \tag{4.4}$$

$$P(w_i|z_i{=}j,\phi_j) = \phi_{w_{i,j}} \tag{4.5}$$

$$P(w_i|z_i{=}j,z_{\text{-}i},w_{\text{-}i}){=}Dir(\delta_{i,j}+n_{w_{\text{-}i,j}})\int \phi_{w_{i,j}}d\phi_j \tag{4.6}$$

$$P(w_i|z_i{=}j,z_{\text{-}i},w_{\text{-}i}){=}\frac{n^{w_i}_{\text{-}i,j}+\delta_{i,j}}{n^{(\cdot)}_{\text{-}i,j}+\sum_a^V\delta_{a,j}} \tag{4.7}$$

n^w and n^d in this and the following equations represent a count matrix for the number of times a word is assigned to a topic and the number of times a topic is assigned to a document respectively. For brevity since the prior probability is unchanged in the “Bijective Mapping” model we will skip the derivation which is well defined in other articles [GS04, Dar11, Gri02]. The derived prior equation is given as:

$$P(z_i{=}j|z_{\text{-}i}){=}\frac{n^{d_i}_{\text{-}i,j}+\alpha}{n^{(d_i)}_{\text{-}i}+K\alpha} \tag{4.8}$$

Putting the two equations together gives the final Gibbs sampling equation:

$$P(z_i{=}j|z_{\text{-}i},w) \propto \frac{n^{w_i}_{\text{-}i,j}+\delta_{i,j}}{n^{(\cdot)}_{\text{-}i,j}+\sum_a^V\delta_{a,j}}\frac{n^{d_i}_{\text{-}i,j}+\alpha}{n^{(d_i)}_{\text{-}i}+K\alpha} \tag{4.9}$$

Given the approximation to the topic assignments, the θ and ϕ distributions are calculated as:

$$\phi_{w,t} = \frac{n_{w,t}+\delta_{w,t}}{n_t+\sum_a^V\delta_{a,t}} \tag{4.10}$$

$$\theta_{t,d} = \frac{n_{d,t}+\alpha}{n_d+K\alpha} \tag{4.11}$$

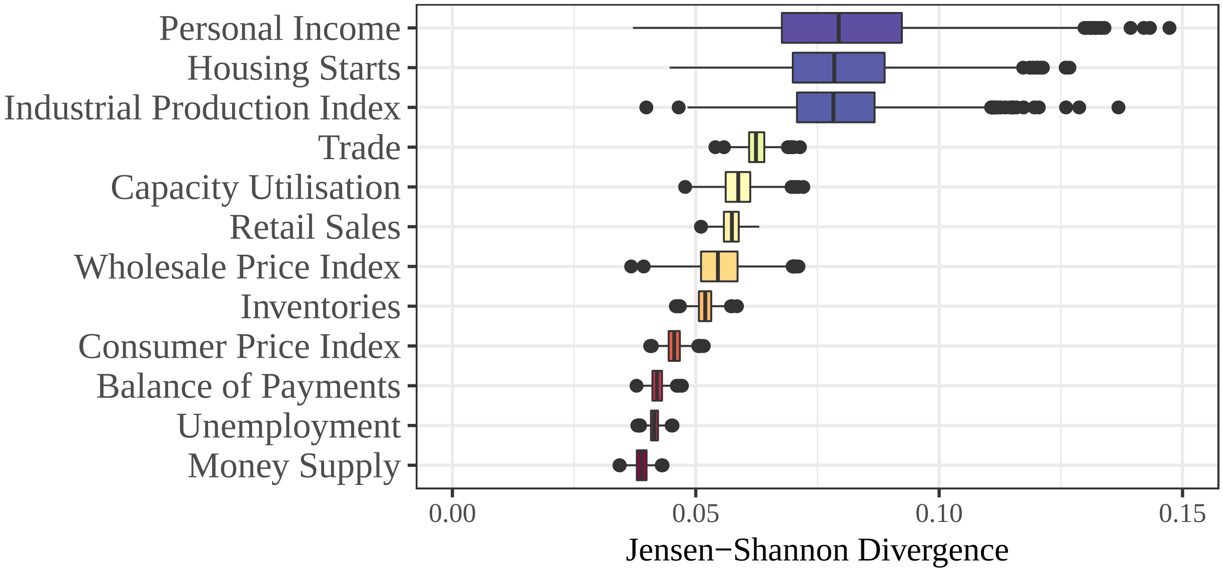


Figure 4.2: Box plots [Bec14] showing the Jensen-Shannon divergence (the JS divergence measures the distance or similarity between probability distributions) of 1000 Dirichlet samples parameterized by source hyperparameters for a subset of knowledge source topics. The topics were derived from Wikipedia pages.

In the case when all topics are known, this model has the advantage of conforming the ϕ distributions to the source distributions, but has three drawbacks. First, even though there is some variability between the ϕ distribution and source distribution, as illustrated by Figure 4.2, there may be cases in which this constraint should be relaxed even further. This is because it is entirely possible to generate a corpus about a known topic without exactly following the frequencies at which the topic is discussed in its respective knowledge source article. This model also requires the user to input the known labeled topics, and other possible supervised approaches may be better suited to the task [BM07, LSJ08, RHN09]. The third drawback is that we are not allowing the possibility that the corpus was generated from a mixture of known topics (labeled topics influenced by a knowledge source topic) and unknown topics, which is a more realistic scenario for an arbitrary corpus. The next model aims to resolve this last deficiency.

4.1.2 Known Mixture of Topics

The next model assumes the topic model is given how many topics are known topics (as well as their word distributions) and how many are unknown topics. The previous approach (bijective mapping) works quite well in this situation in that an unknown topic will have a symmetric beta parameter which will capture assignments which were unallocated due to a low probability in matching any known topic.

The resulting model helps to solve the existing problems of the bijective model and only requires a minor input to the bijective model's generative model. The model changes as shown below with a minor change to the generative algorithm and the collapsed Gibbs sampling.

```
1.  For each of the $K$ topics $\phi_k$:
2.      **if** $k \leq T$ **then**
3.          Choose $\phi_k \sim \text{Dir}(\beta)$
4.      **else**
5.          $\delta_k \leftarrow (X_{k,1}, X_{k,2}, \ldots, X_{k,V})$
6.          Choose $\phi_k \sim \text{Dir}(\delta_k)$
```

Where T is the total number of non-source (short for non knowledge source) topics. The change required to the collapsed Gibbs sampling is then:

$$P(z_i{=}j|z_{\text{-}i},w) \propto \frac{n^{w_i}_{\text{-}i,j} + \beta}{n^{(\cdot)}_{\text{-}i,j} + W\beta} \frac{n^{d_i}_{\text{-}i,j} + \alpha}{n^{(d_i)}_{\text{-}i} + K\alpha}, \ \forall i \leq T \tag{4.12}$$

and

$$P(z_i{=}j|z_{\text{-}i},w) \propto \frac{n^{w_i}_{\text{-}i,j} + \delta_{i,j}}{n^{(\cdot)}_{\text{-}i,j} + \sum_a^V \delta_{a,j}} \frac{n^{d_i}_{\text{-}i,j} + \alpha}{n^{(d_i)}_{\text{-}i} + K\alpha}, \ \forall i > T \tag{4.13}$$

This approach gives the benefit of allowing a mixture of known topics and unknown topics, but problems still arise in that the Dirichlet distributions for the source distribution may be too restricting.

4.1.3 Unkown Mixture of Topics

By using the counts as hyperparameters, the resultant ϕ distribution will take on the shape of the word distribution derived from the knowledge source. However, this might be at odds with the aim of enhancing existing topic modeling. With the goal to influence the ϕ distribution, it is entirely plausible to have divergence between the two distributions. In other words, ϕ may not need to strictly follow the corresponding knowledge source distribution.

4.1.3.1 Variance from the source distribution

To allow for this relaxation, another parameter λ is introduced into the model which is used to allow for a higher deviance from the source (short for knowledge source) distribution. To obtain this variance each source hyperparameter will be raised to a power of λ. Thus, as λ approaches 0, each hyperparameter will approach 1 and the subsequent Dirichlet draw will support all discrete distributions with equal probability. As λ approaches 1, the Dirichlet draw will be tightly conformed to the source distribution.

The addition of λ changes the existing generative model only slightly and allows for a variance for each individual δ_i—which frees us from an overly restrictive binding to the associated knowledge source distribution. The λ parameter acts as a measure of how much divergence is allowed for a given modeled topic from the knowledge source distribution. Figure 4.3 shows how the Jensen-Shannon (JS) Divergence changes with changes to the λ parameter. The change needed to the general model is given as:

5. $$\delta_k \leftarrow [(X_{k,1})^\lambda, (X_{k,2})^\lambda, \ldots, (X_{k,V})^\lambda]$$

With the introduction of λ as an input parameter, the new topic model has the advantage of allowing variance and also leaves the collapsed Gibbs sampling equation unchanged. However, this also requires a uniform variance from the knowledge base distribution for all latent topics. This can be a problem if the corpus was generated with some topics influenced strongly while others less so. To solve this we can introduce λ as a hidden parameter of the model.

4.1.3.2 Approximating λ

In the ideal situation, λ will be as close to 1 for most knowledge source based latent topics, with the flexibility to deviate as required by the data. For this we assume a Gaussian prior over λ with the mean set to μ. The variance then becomes a modeled parameter that conceptually can be thought of as: how much variance from the knowledge source distribution we wish to allow in our topic model. In assuming a Gaussian prior for λ, we must integrate λ out of the collapsed Gibbs sampling equations (only the probability of w_i under topic j is shown, the probability of topic j in document d is unchanged and omitted).

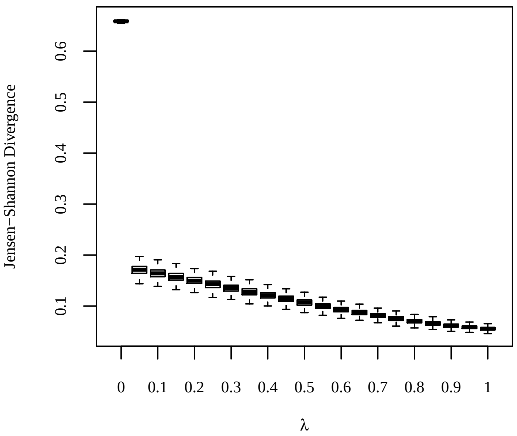


Figure 4.3: Box plots showing how the JS divergence between a source distribution and a Dirichlet sample parameterized by source hyperparameters raised to λ changes with changes to λ without smoothing.

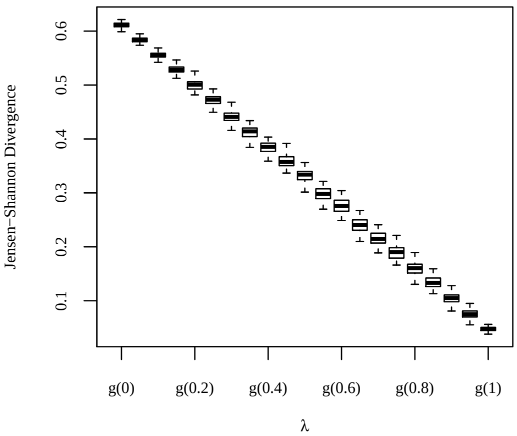


Figure 4.4: The JS divergence between a source distribution and a Dirichlet sample parameterized by source hyperparameters raised to λ with λ mapped to a linear smoothing function g.

$$P(z_i{=}j|z_{\text{-}i},w) \propto \int \frac{n^{w_i}_{\text{-}i,j} + (\delta_{i,j})^{\lambda}}{n^{(\cdot)}_{\text{-}i,j} + \sum_a^V (\delta_{a,j})^{\lambda}} \mathcal{N}(\mu,\sigma) d\lambda \tag{4.14}$$

ϕ then becomes

$$\phi_{w,t} = \int \frac{n_{w,t} + (\delta_{w,t})^{\lambda}}{n_t + \sum_a^V (\delta_{a,t})^{\lambda}} \mathcal{N}(\mu,\sigma) d\lambda \tag{4.15}$$

Unfortunately, closed form expressions for these integrals are hard to obtain, so they must be approximated numerically during sampling.

Another problem arises in that the change of λ is not in par with the change of the Gaussian distribution, as can be seen in Figure 4.3. To make the changes of λ more in line with what would be expected from the Gaussian PDF, we must map each individual λ value in the range 0 to 1 with a value which produces a change in the JS divergence in a linear fashion. We approximate a function, $g(x)$ with a linear shape, shown in Figure 4.4. The approach taken to approximate $g(x)$ is by linear interpolation of an aggregated large number of samples for each point taken in the range 0 to 1. Our collapsed Gibbs sampling equations then becomes:

$$P(z_i{=}j|z_{\text{-}i},w) \propto \int \frac{n^{w_i}_{\text{-}i,j} + (\delta_{i,j})^{g(\lambda)}}{n^{(\cdot)}_{\text{-}i,j} + \sum_a^V (\delta_{a,j})^{g(\lambda)}} \mathcal{N}(\mu,\sigma) d\lambda \tag{4.16}$$

$$\phi_{w,t} = \frac{n_{w,t} + \beta}{n_t + V\beta},\ \forall t \leq T \tag{4.17}$$

and

$$\phi_{w,t} = \int \frac{n_{w,t} + (\delta_{w,t})^{g(\lambda)}}{n_t + \sum_a^V (\delta_{a,t})^{g(\lambda)}} \mathcal{N}(\mu, \sigma) d\lambda,\ \forall t > T \tag{4.18}$$

4.1.3.3 Superset Topic Reduction

A third problem involves knowing the right mixture of known topics and unknown topics. It is also entirely possible that many topics derived from the knowledge source may not be used by the generative model. Our desire to leave the model as unsupervised as possible calls for input that is a superset of the actual generative topic selection in order to avoid manual intervention. In the case of modeling only a specific number of topics over the corpus, the problem becomes how to choose which knowledge source influence topics to allow in the model versus how many unlabeled topics to allow.

The goal is to allow for a superset of knowledge source topics as input and then during the inference to select the best subset of these with a mixture of unknown topics where the total number of topics is given as input K. The approach is to initialize a mixture of K unlabeled topics alongside the labeled knowledge source topics. The total number of topics to start the model then becomes T. During inference, we eliminate topics which are not assigned to any documents. At the end of the sampling phase, we can use a clustering algorithm (such as k-means, JS divergence) to further reduce the modeled topics to return a total of K topics. The full collapsed Gibbs sampling algorithm is given in algorithm 1.1.1. The complete generative process is shown in Figure 4.1 and described below:

```
For each of the $T$ topics $\phi_t$:
    if $t \leq K$ then
        Choose $\phi_t \sim \text{Dir}(\beta)$
    else
        Choose $\lambda_t \sim \mathcal{N}(\mu, \sigma)$
        $\delta_t \leftarrow [(X_{t,1})^{g(\lambda_t)}, (X_{t,2})^{g(\lambda_t)}, \ldots, (X_{t,V})^{g(\lambda_t)}]$
        Choose $\phi_t \sim \text{Dir}(\delta_t)$
For each of the $D$ documents $d$:
    Choose $N_d \sim \text{Poisson}(\xi)$
    Choose $\theta_d \sim \text{Dir}(\alpha)$
    For each of the $N_d$ words $w_{n,d}$:
        Choose $z_{n,d} \sim \text{Multinomial}(\theta)$
        Choose $w_{n,d} \sim \text{Multinomial}(\phi_{z_{n,d}})$
```

4.1.3.4 Analysis

By using a clustering algorithm or thresholding the topic document frequency, the collapsed Gibbs algorithm is guaranteed to produce K topics. The running time is a function of the number of iterations I, average words per document D_{avg}, number of documents D, number of initial topics T and number of approximation steps A (from Equation 4.15), and is $\mathcal{O}(I \times D_{\text{avg}} \times D \times T \times A)$. This differs only from the traditional collapsed Gibbs sampling in LDA by an increase of $(T - K)A$. But since we have built the approach to potentially have a large $T - K$ this difference can have a significant impact on running times.

Approaches exist that can parallelize the sampling procedure, but these are often approximations or can potentially have slower than baseline running times [WBS09, NAS07, PNI08]. We present two modifications to the original algorithm that allow for inference while guaranteeing the exactness of the results to the original Gibbs sampling. The first one makes use of prefix sums rules [Ble90] and guarantees a running time of:

$$\mathcal{O}(I \times D_{\text{avg}} \times D \times A \times Max[T/P,P]) \tag{4.19}$$

with P being the number of parallel units. This algorithm is given by Algorithm 1.1.2. This algorithm is practical in situations where $T - K$ is large, but suffers from the limitations of the number of context switches required for the threads to wait at their respective barriers. A simpler implementation approach that reduces the number of context switches is to add the sums for each thread then wait for a barrier. When the barrier is released we add the end values together and then in parallel add the remaining necessary items. This approach is given in Algorithm 1.1.3. The

Algorithm 1.1.1 Collapsed Gibbs

Input: Dirichlet hyperparameters α, β, a corpus C, vocabulary V, unlabeled topic count K, total topic count T, a set of source topics S, mean μ, variance σ, and iteration count I.
Output: θ, ϕ

```
procedure COLLAPSED_GIBBS($\alpha$, $\beta$, $C$, $V$, $T$)
    for $t = K + 1$ to $T$ do
        Calculate $g_t$
    end for
    Initialize $C_{\text{topics}}$ to random topic assignments
    Update $n^w$ and $n^d$ from $C_{\text{topics}}$
    for $iter = 1$ to $I$ do
        for $i = 1$ to $C$ do
            for $j = 1$ to $|C_i|$ do
                $C_{\text{topics}_{i,j}} \leftarrow Sample(i,j)$
            end for
        end for
    end for
    Calculate $\theta$ according to Equation 1.11
    Calculate $\phi$ according to Equation 1.18
    return $\theta, \phi$
end procedure

procedure SAMPLE($i$, $j$)
    Decrement $n^w$ and $n^d$ accordingly
    for $t = 1$ to $K$ do
        Calculate $p_t$ according to Equation 1.13
    end for
    for $t = K + 1$ to $T$ do
        Calculate $p_t$ according to Equation 1.16
    end for
    $topic \sim$ Multinomial($p$)
    Increment $n^w$ and $n^d$ accordingly
    return $topic$
end procedure
```

running time remains the same as the prefix sums running time as:

$$\mathcal{O}(I \times D_{\text{avg}} \times D \times A \times Max[T/P,P]) \tag{4.20}$$

These two algorithms allow for mitigation of the increase in the number of topics and should approach running times similar to those of standard LDA runs. They are also very extensible and can be used in other optimization algorithms.

4.1.3.5 Input determination

Determining the necessary parameters and inputs into LDA is an established research area [WMS09], but since the proposed model introduces additional input requirements, a brief overview will be given about how to best set the parameters and determine the knowledge source.

Algorithm 1.1.2 Prefix Sums Parallel Sampling

```
procedure SAMPLE($i, j$)
    Decrement $n^w$ and $n^d$ accordingly
    for $i$ from 0 to $T - 1$ in parallel do
        if $i \leq K$ then
            Calculate $p_i$ according to Equation 1.13
        else
            Calculate $p_i$ according to Equation 1.16
        end if
        $p_i \leftarrow p_{i-1} + p_i$
    end for
    for $d$ from 0 to $(\ln T) - 1$ do
        for $i$ from 0 to $T - 1$ by $2^{d+1}$ in parallel do
            $p_{(i+2^{d+1}-1)} \leftarrow p_{(i+2^d-1)} + p_{(i+2^{d+1}-1)}$
        end for
    end for
    $p_{(T-1)} \leftarrow 0$
    for $d$ from $(\ln T) - 1$ down to 0 do
        for $i$ from 0 to $T - 1$ by $2^{d+1}$ in parallel do
            $h \leftarrow p_{(i+2^d-1)}$
            $p_{(i+2^{d+1}-1)} \leftarrow p_{(i+2^{d+1}-1)}$
            $p_{(i+2^{d+1}-1)} \leftarrow h + p_{(i+2^{d+1}-1)}$
        end for
    end for
    $topic \leftarrow$ Binary_Search($p$)
    Increment $n^w$ and $n^d$ accordingly
    return $topic$
end procedure
```

Algorithm 1.1.3 Simple Parallel Sampling

```
procedure SAMPLE($i, j$)
    Decrement $n^w$ and $n^d$ accordingly
    for $i$ from 0 to $T - 1$ in parallel do
        if $i \leq K$ then
            Calculate $p_i$ according to Equation 1.13
        else
            Calculate $p_i$ according to Equation 1.16
        end if
        $p_i \leftarrow p_{i-1} + p_i$
    end for
    for $i$ from 0 to $T - 1$ by $T/P$ do
        $p_i \leftarrow p_{(i-T/P)} + p_i$
        $ends_i \leftarrow p_i$
    end for
    for $i$ from 0 to $T - 1$ in parallel do
        $diff \leftarrow p_{end} - ends_i$
        $p_i \leftarrow diff + p_i$
    end for
    $topic \leftarrow$ Binary_Search($p$)
    Increment $n^w$ and $n^d$ accordingly
    return $topic$
end procedure
```

4.1.3.5.1 Parameter selection

To determine the appropriate parameters, techniques utilizing log likelihood have previously been established [GS04]. Since these approaches generally require held out data and are a function of the ϕ, θ, and α variables, the introduction of λ and σ will not differentiate from their original equations. For example, the perplexity calculations used for Source-LDA are based off importance sampling [WMS09], or latent variable estimation via Gibbs sampling [Hei08]. Importance sampling is only a function of ϕ given by Equation 4.18, and estimation via Gibbs sampling can made using Equation 4.18 and the following equation ($\tilde{z}$, $\tilde{w}$, and $\tilde{n}$ represent the corresponding variables in the test document set):

$$P(\tilde{z}_i{=}j|\tilde{z}_{\text{-}i},\tilde{w}) \propto \frac{n_j^{w_i} + \tilde{n}_{\text{-}i,j}^{w_i} + \beta}{n_j^{(\cdot)} + \tilde{n}_{\text{-}i,j}^{(\cdot)} + W\beta} \frac{\tilde{n}_{\text{-}i,j}^{d_i} + \alpha}{\tilde{n}_{\text{-}i}^{(d_i)} + K\alpha}, \ \forall i \leq T \tag{4.21}$$

and

$$P(\tilde{z}_i{=}j|\tilde{z}_{\text{-}i},\tilde{w}) \propto \frac{n_j^{w_i} + \tilde{n}_{\text{-}i,j}^{w_i} + \delta_{i,j}}{n_j^{(\cdot)} + \tilde{n}_{\text{-}i,j}^{(\cdot)} + \sum\limits_{a}^{V} \delta_{a,j}} \frac{\tilde{n}_{\text{-}i,j}^{d_i} + \alpha}{\tilde{n}_{\text{-}i}^{(d_i)} + K\alpha}, \forall i > T \tag{4.22}$$

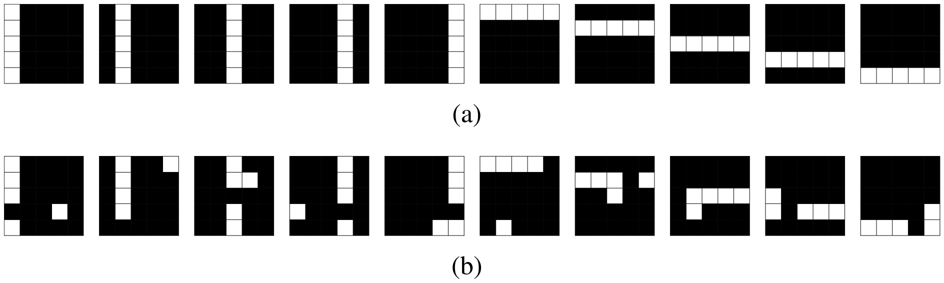
It is recommended to set the parameters so as to maximize the log likelihood. Further analysis such as whether or not the parameters can be learned a priori from the data are not the focus of this work and are thus left as an open research area.

4.1.3.5.2 Knowledge source selection

Source-LDA is designed to be used only with a corpus which has a known super set of topics which comprise a large portion of the tokens. An example of such a case is that of a corpus consisting of clinical patient notes. Since there are extensive knowledge sources comprising essentially all medical topics, Source-LDA can be useful in discovering and labeling corpora from this domain. In cases where it is not so easy to collect a superset of topics, traditional approaches may be more useful.

4.1.4 Evaluation

To test the results of the Source-LDA algorithm, we set up experiments to test against competing models. The most similar models to our proposed approach were used in comparison. These are: latent Dirichlet allocation (LDA) [BNJ01], explicit Dirichlet allocation (EDA) [HRS13], and the Concept-topic model (CTM) [SSC11]. Other approaches such as supervised latent Dirichlet allocation (sLDA) [BM07], discriminative LDA (DiscLDA) [LSJ08], and labeled LDA (L-LDA) [RHN09]



(a)

(b)

Figure 4.5: A graphical representation of topics containing 1 word for the cell locations of row and column vectors in a 5 x 5 picture (a) and their augmented topics after swapping a random assigned word (pixel) with a random topic's assigned word (b).

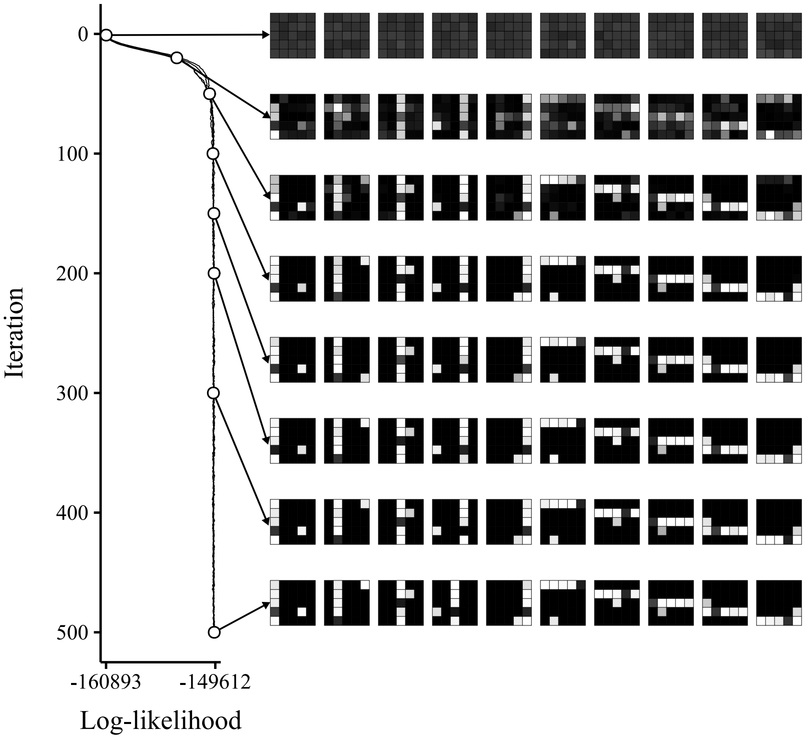


Figure 4.6: Results from running Source-LDA for a corpus generated from topics in Figure 4.5(b) using a knowledge source of topics corresponding to Figure 4.5(a). Four separate runs are plotted to show the similarity of the log-likelihood relation to the iteration between the runs. The topics are shown visually at iteration 1, 20, 50, 100, 150, 200, 300 and 500 for a single run.

are not used since a main desideratum of Source-LDA is to require much less supervision than what is needed by these methods. Likewise, hierarchical methods [KML12] are omitted because there is no established hierarchy in the knowledge source data for this model. We describe in more detail below the experimental setups and metrics used to compare results.

4.1.4.1 A Graphical Example

Following a previously established experiment [GS04], we show the utility of Source-LDA by visualizing topics created with words that correspond to the pixel locations in a 5×5 picture—but we add a key difference. The original topics are augmented, used to generate a corpus, and then hidden. Only the non augmented topics are given as input with the goal of discovering the augmented topics using the corpus and their original topics.

4.1.4.1.1 Experimental Setup

We start by creating ten topics with the vocabulary being the set of pixel locations in a 5×5 picture. The vocabulary (V) and bag of words representation of a topic (T_i) are defined as:

$$V = \{xy \mid 0 \leq x < 5 \wedge 0 \leq y < 5\}$$

$$T_i = \begin{cases} xy \mid y = i \wedge 0 \leq x < 5, & \text{if } 0 \leq i < 5 \\ yx \mid y = i \wedge 0 \leq x < 5, & \text{otherwise} \end{cases}$$

The topics are shown by Figure 5(a) with the intensity (I) of a pixel corresponding to word w in topic t equal to:

$$I(w, t) = Max[5 \times P(w|t), 1]$$

The representation of topics in this manner leads to a total of 10 topics. These original topics are then augmented by pairing each topic with a random different topic and swapping a random word (pixel). Figure 4.5(b) shows the augmented topics which represent a 20% augmentation rate between the original topics. From the set of augmented topics, we generate a 2,000 document corpus using the generative model of LDA. Each document consists of 25 words with topic assignments drawn from a distribution sampled from the Dirichlet distribution parameterized by $\alpha = 1$. With the knowledge source consisting solely of the original non augmented topics, we run Source-LDA on the corpus hoping to discover and properly label the augmented topics. For comparative analysis we also run EDA and CTM against the same data set.

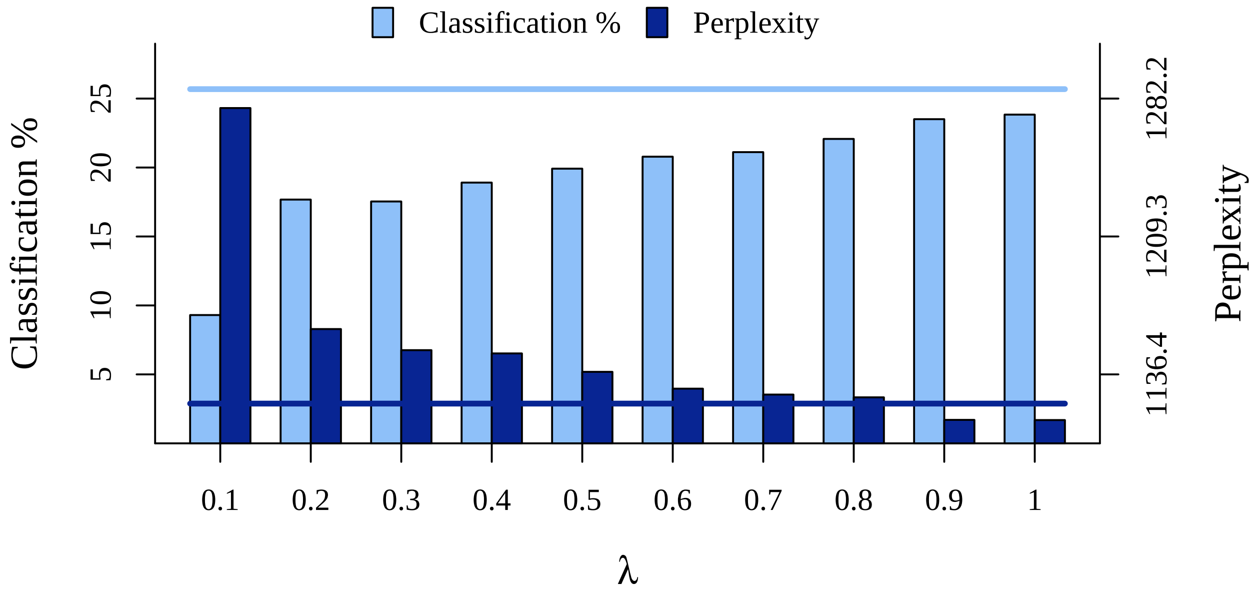


Figure 4.7: Classification accuracy and perplexity values for fixed values of λ compared against the baseline values generated from a dynamic λ with a normal prior. The baseline values shown as lines represent the classification percentage of 25.7 and perplexity value of 1119.9

4.1.4.1.2 Experimental Results

As shown in Figure 4.6, Source-LDA discovers the augmented topics given the set of original topics. Not only is Source-LDA able to find the topics correctly to the augmented distributions used in the generation of the corpus, but it is also able to match them to their respective non augmented knowledge source distributions. This simple experiment highlights a big advantage of Source-LDA—which is the ability to discover topics that differ from their respective weakly-supervised input set. Other models such as EDA and CTM are unable to label the augmented topics correctly due to the topics containing a word (pixel) not in the original distribution. The comparative average JS divergence was 0.012, 0.138, and 0.43 for Source-LDA, EDA, and CTM respectively.

4.1.4.2 Integrating λ

A reasonable assumption of a corpus in which some topics are generated from a knowledge source is that the topics used in the corpus are going to deviate (more or less similar) from their respective knowledge source distributions, and that each individual topic is going to deviate at a different rate than other topics. The introduction of λ to Source-LDA as a parameter to be learned by the data allows the flexibility of different topics to be influenced differently by λ, but comes at an increase in computation cost. To show that in certain cases this flexibility is needed to obtain more accurate results, we derive an experiment consisting of topics with different deviations from their respective source distributions.

4.1.4.2.1 Experimental Setup

A synthetic 500 document corpus is generated from a knowledge source of 100 randomly selected Wikipedia topics. The corpus is generated using the bijective model of Source-LDA as outlined in Section 4.1.1, consisting of 100 topics, an average word count per document of 100 words, $\mu = 0.5$, $\sigma = 1.0$ and $\alpha = 0.5$. Furthermore, even though for each topic λ was drawn from $\mathcal{N}(\mu, \sigma^2)$, we bound the value drawn to the interval $[0,1]$ for comparative analysis. We then run Source-LDA under the bijective model for a baseline of $\mu = 0.5$, $\sigma = 1.0$ against 10 runs of Source-LDA with λ fixed. After each run we compare the classification accuracy and perplexity values.

4.1.4.2.2 Experimental Results

For all fixed λ runs the baseline approach of varying λ in accordance with the normal distribution results in a higher classification accuracy. By allowing λ to deviate, the model can make up for less accurate parameter determination based on maximizing perplexity. As shown in Figure 4.7, classification accuracy is not perfectly correlated with perplexity. This is shown by the baseline method reporting a higher perplexity value than the fixed $\lambda = 1$ value while maintaining a higher classification accuracy. Even though we still recommend perplexity or other log-likelihood maximization approaches to set the parameters in any unknown data set, maximizing log-likelihood has been shown to be a less than perfect metric for evaluating topic models [CBG09, AOC16]. In this experiment and the remaining experiments we take classification accuracy to be a more appropriate measurement for evaluating topic models.

	Source-LDA	IR-LDA	Concept Topic Model
Inventories			
	inventory	systems	sales
	cost	products	year
	stock	said	sold
	accounting	information	retail
	goods	technology	given
	management	company	place
	time	data	marketing
	costs	network	improved
	financial	kodak	passed
	process	available	addition
Natural Gas			
	gas	corp	gas
	natural	contract	said
	used	company	total
	water	services	value
	oil	unit	near
	carbon	subsidiary	natural
	cubic	completed	properties
	energy	work	california
	fuel	dlr	wells
	million	received	future
Balance of Payments			
	account	said	said
	surplus	public	june
	deficit	state	april
	current	private	beginning
	balance	planned	great
	currency	reduce	later
	trade	local	remain
	exchange	added	reserve
	capital	make	equivalent
	foreign	did	imported

Table 4.1: Topics and their most probable word lists for Source-LDA, IR-LDA, and CTM.

4.1.4.3 Reuters Newswire Analysis

To show the type of topics discovered from Source-LDA, we run the model on an existing dataset. This collection contains documents from the Reuters newswire from 1987. The dataset contains 21,578 articles, spanning a large set of categories. One important feature of the dataset is a set of given categories that we can use for our topic labeling. These include broad categories such as shipping, interest rates, and trade, as well as more refined categories such as rubber, zinc, and coffee. Our choice to apply Source-LDA to this dataset is due to the fact that the Reuters dataset is widely used for information retrieval and text categorization applications. Due to its widespread use, it can considerably aid us in comparing our results to other studies. Additionally, because it contains distinct categories that we can use as our known set of topics, we can easily demonstrate the viability of our model.

4.1.4.3.1 Experimental Setup

Source-LDA, LDA, and CTM were run against the Reuters-21578 newswire collection. Since EDA does not discover new topics, nor does it update the word distributions of the input topics, we did not include EDA in this experiment. From the original 21,578 document corpus we select a subset of 2,000 documents. The Source-LDA and CTM supplementary distributions were generated by first obtaining a list of topics from the Reuters-21578 dataset. Next, for each topic, the corresponding Wikipedia article was crawled and the words in the topic were counted, forming their respective distributions (counts for each word divided by the total word count). Querying Wikipedia resulted in 80 distinct topics as our superset for the knowledge source. Out of the 80 crawled available topics, only 49 topics appear in the 2,000 document corpus. This represents the ideal conditions in which Source-LDA is to be applied—that of a corpus in which a significant portion of tokens are generated from a subset of a larger and relatively easy to obtain topic set. For all models, a symmetric Dirichlet parameter of $50/T$ (where T is the number of topics) and $200/V$ (where V is the size of the vocabulary) was used for α and β respectively. For Source-LDA, μ and σ were determined by experimentally finding a local minimum value of perplexity, which resulted from the parameter values of 0.7 for μ and 0.3 for σ. The bag of words used in the CTM were taken from the top 10,000 words by frequency for each topic. The models showed good convergence after 1,000 iterations, so the number of iterations parameter was set to 1,000. After sampling was complete for LDA, the resulting topic-to-word distribution was mapped using an information retrieval (IR) approach. The IR approach was to use cosine similarity of documents mapped to term frequency-inverse document frequency (TF-IDF) vectors with the TF-IDF weighted query vectors formed from the top 10 words per topic.

4.1.4.3.2 Experimental Results

After the LDA model converged, we label the topics using the IR approach described above (we referred to this topic labeling method as IR-LDA). Given similar labels from the models, it is an intuitive approach to compare the word assignments of each topic model. Example comparisons are shown in Table 4.1. The label assignments generated from Source-LDA show a more accurate assignment of labels to topics than both IR-LDA and CTM. IR-LDA appears to suffer from the mixing of different concepts into a single topic, for example with the topic "Inventories," the topic assignments could possibly be the combination of "Inventories" and "Information Technology". The CTM seems to assign more weight to less important words. One approach to rectify this problem is to use a smaller number of words for the bag of words, but this leads to significant dropout and no labeled topics are passed through. Out of the total 100 returned topics, CTM only discovered 6 labeled topics, with Source-LDA discovering 15. Since the IR approach forces all topics to a label regardless of the quality of the label, IR-LDA returned 100 labeled topics. Out of the 6 labeled CTM topics only 3 were overlapping with Source-LDA and IR-LDA and are shown in Table 4.1. The remaining 3 CTM topics were bad matches for the label with an average of 86% of words not appropriate for the label as determined by human judgment (we acknowledge the potential for bias). Meanwhile Source-LDA mismatched at a rate of 36%, with IR-LDA at a rate of 77%. The top words from topics discovered by Source-LDA are more consistent with the meaning of the topic as opposed to what words you may find in a topic discovered by LDA, which can be generally applied to many concepts.

4.1.4.4 Wikipedia Corpus

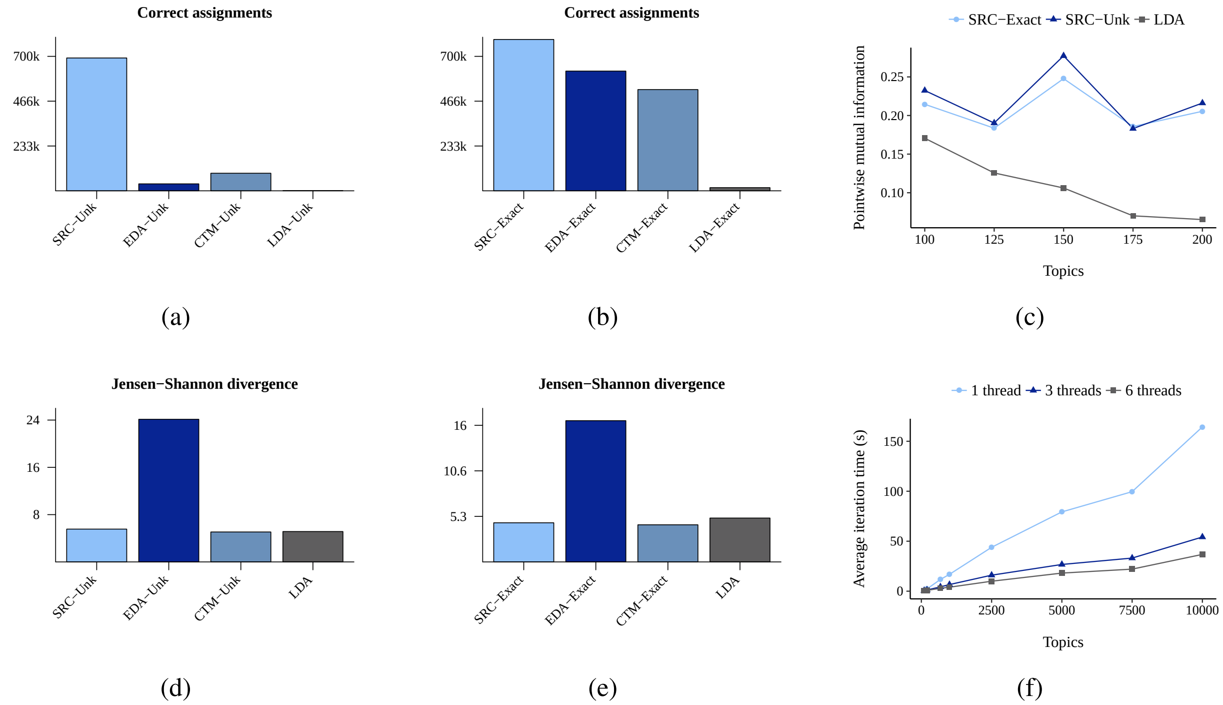
A comparison of Source-LDA against EDA, and CTM is made using a corpus generated using a known knowledge source corresponding to medical topics extracted from MedlinePlus[1] (a consumer-friendly medical dictionary). We evaluate the strength of Source-LDA under different models proposed in Section 4.1 using the metrics of classification accuracy, JS divergence and Pointwise mutual information (PMI).

PMI is an established evaluation of learned topics which takes as input a subset of the most popular tokens comprising a topic and determines the frequency of all pairs in the subset occurring at a given input distance from each other in the corpus. The more that these pairs occur close to each other then the better the learned topics. PMI differs from the JS divergence evaluation for this experiment in that PMI will tell us how good our topics are, where as the JS divergence will tell us how good our distribution over topics for each document is.

4.1.4.4.1 Experimental Setup

A corpus of Wikipedia vocabulary articles was generated by following the steps of the generative model for Source-LDA, where the chosen K topics are a subset of a larger collection of Wikipedia topics (topics formed from Wikipedia articles). The topics consisted of 578 Wikipedia articles representing corresponding articles from MedlinePlus. The number of topics (K) was given as 100, chosen from the entire collection of 578 topics (B), the number of documents (D) was given as 2000 and the average document word count (D_{avg}) as 500, μ and σ were set to 5.0 and 2.0 for the bijective evaluation, and 0.7 and 0.3 for the Source-LDA model respectively. After these 2000 documents were generated, the topic assignments were recorded and used as the ground truth measurement. The first round of topic models consisted of comparing Source-LDA, EDA, and CTM. For Source-LDA μ and σ were set to match that of the generative model. For all models, a symmetric Dirichlet parameter of $50/T$ and $200/V$ was used for α and β respectively. After convergence of the models they were evaluated against the ground truth measurement. In the second round of experiments each

[1] https://www.nlm.nih.gov/medlineplus/



(a) (b) (c)

(d) (e) (f)

Figure 4.8: Results showing the number of correct topic assignments in the mixed model (a) and bijective model (b) and sum total of the JS divergences of θ in the mixed (d) and bijective models (e). Sorted PMI analysis for a Wikipedia generated corpus inferred by the exact bijective model and mixed model is shown by (c). Performance benchmarking is given in (f).

topic model was run under the bijective model, that is they only considered topics which were used in the ground truth assignments.

To compare Source-LDA against LDA using PMI, 5 corpora were generated under the bijective model with the number of topics K ranging from 100 to 200. B, D, D_{avg}, μ, and σ were set to 100, 578, 200, 300, 1.0 and 0.0 respectively. The parameters for Source-LDA followed the generative model and all other parameters are the same as the previous experiments. After 1000 iterations the top 10 words given for each topic were used in the PMI assessment.

4.1.4.4.2 Experimental Results

The topic assignments for each token in the corpus were recorded for all models and the results compared against each other. Since we know a priori the correct topic assignment for each token, we use the number of correct topic assignments to be an appropriate measure of classification accuracy. Note that in evaluations where the ground truth is known, classification accuracy is a much better determination of the goodness of a model than log likelihood maximizations such as perplexity and therefore we do not evaluate the model using perplexity. In Figure 4.8, all topic models run under the full Source-LDA (mixed) model are tagged with an "Unk" label, and likewise topic models run under the bijective model are tagged with "Exact". The overall number of correct topic assignments for each model are shown in Figure 4.8(a) for the mixed model and Figure 4.8(b) for the bijective model. Since the LDA model has unknown topics, JS divergence was used to map each LDA topic to its best matching Wikipedia topic. As expected the Source-LDA model (SRC-Unk and SRC-Exact) had the best results amongst all other topic models for classification accuracy.

In the second analysis the topic to document distributions were analyzed using sorted JS Divergence, and is irrespective of any unknown mapping. The results again show the Source-LDA model to be effective in accurately mapping topics to documents in both the case where the topics used in the generative model are unknown (Figure 4.8[d]) and where the topics are known (Figure 4.8[e]). Even though an accurate alignment of θ by itself does not lend much weight to any one model being superior, we do find it important to demonstrate how θ is being affected by the different algorithms.

The PMI analysis detailed by Figure 4.8(c) show that by PMI, Source-LDA provides a better mapping of labels to topics over the input corpora. This is an encouraging result, even though the differences are not large, since LDA is a function of topic proximity in a document and word frequency in a topic, whereas Source-LDA is a function of the same plus the likelihood of a word being in an augmented source distribution.

4.1.4.5 Performance Benchmarking

To show the performance gains used by the parallel sampling algorithm an experiment was set up to generate topics randomly from a given vocabulary. The corpus was generated using the same parameters as in Section 4.1.4.2, but with B ranging from 100 to 10000. The benchmarking is visualized by Figure 4.8(d). It clearly demonstrates that Source-LDA is linearly scalable and easily parallelized.

4.1.5 Discussion

Source-LDA represents a novel methodology for weakly-supervised topic modeling to discover labeled interpretable topics. Additionally, this work provides parallel algorithms to speed up the inference process. This methodology uses prior knowledge sources to influence a topic model in order to allow the labels from these external sources to be used for topics generated over a corpus of interest. In addition, this approach results in more interpretable topics generated based on the quality of the external knowledge source. We have tested our methodology against the Reuters-21578 newswire collection corpus for labeling and Wikipedia as an external knowledge source. The analysis of the quality of topic models using PMI show the ability of Source-LDA to enhance existing topic modeling interpretability.

4.2 ReSource-LDA

Source-LDA represents a pioneering and novel approach to labeling interpretable topics in a single topic model, however it is not infallible. One such drawback is represented by n-grams which do not belong to the input knowledge source. For these sets of tokens, the model assumes a token assignment that is uniform across knowledge source topics. In the event a token from a corpus does not exist in the knowledge source, it may be advantageous to leverage neural networks to detect whether the unseen word linguistically resembles the existing tokens that make up a particular knowledge source topic.

Deep neural networks have been established as an effective technique to pattern recognition and machine learning [dee12, Kri12, CW08]. A subclass of these models is the recurrent neural network (RNN), which utilizes previous states in training of the current state. An interesting effect to training these models is the ability to generate sequences of data on a diverse range of input. The RNN is particularly interesting in its ability to capture and generate the appropriate context. This is due to the model in which it is trained. For each time-step the hidden layer is influenced by the input axons as well as axons from previous hidden layers. This allows the model to learn not only the current pattern, but previous occurrences as well. In the generation phase, it is easy to predict the next output by sampling from the model's predictive distribution. A series of predictions can then be chained together giving an entire sequence of predicted output. When put all together, the stochastic sampling produces output which is often referred to as a product of dreaming.

Although there are applications that utilize these generated sequences [Gra13, War00, SZ14, GDG15] they often rely on isolated sequences. The ability to connect sequences would have tremendous utility in areas such as anonymization of private data where a foolproof way to guarantee anonymization is to generate a representation of the data that does not share any of the original data.

This however is not a trivial task. In this paper, we work towards the goal of incorporating context around generated segments by integrating into the RNN model information acquired from topic modeling. Because topic modeling is most naturally used in text, we restrict the methods to those applied in text—and because we are interested in applications where a new word is generated, we focus on character level generation.

Given topic models are used at the word level it may be a natural inclination to seek a combination with a word-based RNN. However, we are interested in utilizing the RNN for words that do not belong to a given input vocabulary (such as anonymization of patient data). A word-based RNN would simply not work. Additionally, word-based RNN's can be problematic due to a large input size. For input layers such as one-hot encoding the size of each input equals the size of the vocabulary. For large corpora this can degrade computation time and predictive power [Gra13].

This directionality is not only limited to the topic model influencing the RNN. Just as the ability of topic modeling features used in the RNN helps to add context and thus improve performance, performance gains can also be realized by adding the RNN model into the topic modeling process. The second part of this work investigates the utility of adding a character level RNN into a previously established topic model.

Probabilistic topic models have long been established as an effective means of discovering underlying semantic themes in a set of input [BNJ03, GS04, RGS04]. The current state of the art topic models are derived from latent Dirichlet allocation (LDA) [BNJ03]. LDA makes a simplifying assumption about how a corpus is generated. Since it is too hard to mathematically model the precise steps a human generates a corpus, LDA assumes that words and topics are selected from sampling of distributions and repeated for the length of the corpus.

Traditional LDA suffers from the problem of topic interpretability [SSC11]. That is, given the output from the topic model, by just looking at the topic to word distributions, it is often hard to identify a single n-gram for which this topic is referring to. To help mitigate this problem different approaches have been proposed which either label the topic with a best guess after topic modeling [MSZ07, Mei06, MZ05, MZ06], or as we show in Section 4.1, somehow incorporate existing information into the topic model during inference [WTW17, Han13, SSC11].

To the best of our knowledge, no one has yet to leverage RNNs into the topic modeling process to help in topic interpretability, topic labeling, and potentially improve the overall perplexity of a given topic model—existing work focuses on improving the non-labeled topics [DWG17]. Given that the RNN is a supervised model, and there exist weakly-supervised topic models aimed at identifying and aiding traditional topic modeling, it is an intuitive approach to add the RNN into the weakly-supervised part of the topic model. By doing this, relevant context can be applied that can enhance the existing weakly-supervised portion of the topic model. Our approach is to train the weakly-supervised input with an RNN and use the RNN's predictive power to influence the probability of a given topic assignment to a token during inference.

While there continues to be progress in other approaches of text mining, such as the Transformer [DCL19], or convolutional neural network (CNN) based approaches [Luo19], the RNN is still incredibly important. The authors estimate that RNNs comprise 25% of current deep learning publications (based on a Google Scholar search for papers published in 2019). Additionally, our approach is not only limited to text mining. We envision our technique of combining topic modeling to RNNs to be useful in data sets used with RNNs where topic models form meaningful clusters.

This work also contributes to the sub-fields of topic labeling, topic interpretability and their applications. For example, improving the labels of topics can result in a better understanding of protein function [Liu17], aid the synthesis of data for biologists [Liu16], and help end-users browse large textual databases [Vel]. Aside from the known applications of topic labeling, the theoretical possibilities are equal as important. One such application would be the automatic labeling of patient records. The labels would act as summarizations which could then be given to primary care physicians who are faced with too much information to process in not enough time [Mar06].

4.2.1 Approach

The ability to combine RNNs and topic models can result in a beneficial outcome for both, but the best approach is not so intuitive while maintaining the restriction of using a character level RNN. For consistency in measurement, we choose the same parameters for all models. For all RNNs we choose maximum likelihood estimation for the loss function and backpropagation through time (BPTT) [Wer90] as the technique to minimize loss. We explore different methods of adding the topic modeling information into the RNN. Each technique is illustrated in Figure 4.9. We then describe in more detail our method to add the RNN into the weakly-supervised topic model, Source-LDA, to improve upon the labeling of topics and topic interpretability.

4.2.1.1 RNNs with topic models

4.2.1.1.1 Topic input vocabulary

In the simplest approach, we employ a character-level RNN to generate character sequences (Π-RNN). This is done by appending topic modeling information into the existing input. The change to the existing RNN model is at the input layer. Before training the RNN on the data, we generate a mapping of tokens to topics using an existing topic model. We then take the input set to be the Cartesian product of the character input set and the topic set.

Since the change is only made in the input itself, no differences are made to the corresponding forward and back propagation equations.

4.2.1.1.2 Two-hot encoding

To reduce the input size of the topic input vocabulary approach, it is possible to give as input the character to train as well the topic as a two-hot encoded vector (2-RNN). This serves to shrink the input from size $C \times K$ down to size $C + K$. With C being size of the character vocabulary (Π), and K the total number of topics. In this model the forward and back propagation equations remain unchanged. The execution time and memory are thus reduced from $C \times K$ to $C + K$ in the big $\mathcal{O}$ notation for Π-RNN.

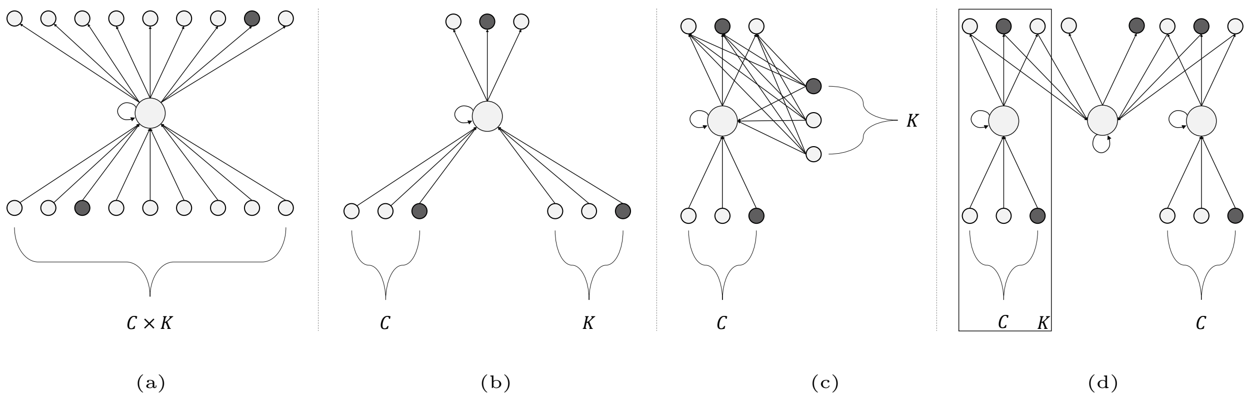


Figure 4.9: Four different ways of combining topic models and RNNs—the first way involves encoding both the topic and character into the input (a), or these input can be added independently to reduce the total input size (b). If we connect the topic layer from (b) to the output layer we can construct the context dependent RNN (c). Lastly, Topic-RNN (d) connects K different character RNNs with one RNN independent of any topics.

4.2.1.1.3 Context dependent

If we interpret the K input layer in the two-hot encoding approach to be independent of the C input layer, and allow a direct influence on the output layer from the K input layer, then we form the basis of the context dependent RNN [MZ12] (CD-RNN). This model includes the influence from the K input layer in both the hidden layer calculations as well as the output layer. Given a sequence of input vectors $(x_1, ..., x_T)$, the RNN predicts the output sequence $(\hat{y}_1, ..., \hat{y}_T)$ using the following equations:

$$\overrightarrow{h_t} = tanh(W^{hx} \times \overrightarrow{x_t} + W^{hh} \times \overrightarrow{h_{t-1}} + W^{hk} \times \overrightarrow{k_t} + \overrightarrow{b^h}) \tag{4.23}$$

$$\overrightarrow{\hat{y}_t} = softmax(W^{yh} \times \overrightarrow{h_t} + W^{yk} \times \overrightarrow{k_t} + \overrightarrow{b^o}) \tag{4.24}$$

Where h_t is the high-dimensional hidden state at the time-step t, W^{hx}, W^{hh}, and W^{yh} are the weight matrices connecting the input to the hidden layer, the previous hidden layer to the current hidden layer, and the hidden layer to the output layer respectively. The vectors b^h and b^o are the biases. The new weights must also be factored into BPTT. The additional updates are:

$$\Delta W^{yk}_{pz} = \eta \times \sum_{t=1}^{T} \delta_{t,p} \times k_{t,z} \tag{4.25}$$

$$\Delta W^{hk}_{jz} = \eta \times \sum_{t=1}^{T} \delta_{t,j} \times k_{t,z} \tag{4.26}$$

With z corresponding to a binary topic input at time t, η is the learning rate and δ are the respective gradients from BPTT. Since we are adding an additional edge set to the output, this increases the time and memory for prediction as $\mathcal{O}(C \times H + K \times H + H^2 + K \times C)$ with H defined as the size of the hidden layer. The training execution time and memory size will be multiplied by the total number of characters in the corpus (D) and the number of time steps to unroll (U) respectively.

4.2.1.1.4 Topic-RNN

While the context dependent RNN effectively uses topics to help the predictive power of the RNN, it is our hypothesis that better results can be achieved by separating topics and their predictions. That is, each topic is given its own RNN and the model overall encompasses these K RNNs. Then the current topic can dictate the entire model's output. Given an additional input of topic assignments, the change needed to the loss function, forward propagations, and backward propagations are to add an extra dimension for each topic in the set of topics as follows:

$$\overrightarrow{h_{t,s}} = tanh(W^{hxs} \times \overrightarrow{x_t} + W^{hhs} \times \overrightarrow{h_{t-1,s}} + \overrightarrow{b^{hs}}) \tag{4.27}$$

$$\overrightarrow{\hat{y}_{t,s}} = softmax(W^{yhs} \times \overrightarrow{h_{t,s}} + \overrightarrow{b^{os}}) \tag{4.28}$$

$$E = -\frac{1}{T} \times \sum_{t=1}^{T} \ln(\hat{y}_{t,s,p}) \tag{4.29}$$

$$\delta_{t,s,p} = -\frac{\partial E}{\partial \hat{y}_{t,s,p}} \times \frac{\partial \hat{y}_{t,s,p}}{net_{t,s,p}} \quad = -\frac{1}{N} \times (1 - \hat{y}_{t,s,p}) \tag{4.30}$$

$$\begin{aligned} \delta_{t,s,j} &= \sum_{p} \delta_{t,s,p} \times W^{yhs}_{pj} \times \frac{\partial h_{t,s,j}}{\partial net_{t,s,j}} \\ &= \sum_{p} \delta_{t,s,p} \times W^{yhs}_{pj} \times (1 - h^2_{t,s,j}) \end{aligned} \tag{4.31}$$

$$
\begin{aligned}
\delta_{t-1,s,m} &= \sum_j \delta_{t,s,j} \times W^{hhs}_{jm} \times \frac{\partial h_{t-1,s,m}}{\partial net_{t-1,s,m}} \\
&= \sum_j \delta_{t,s,j} \times W^{hhs}_{jm} \times (1 - h^2_{t-1,s,m})
\end{aligned}
\tag{4.32}
$$

$$
\Delta W^{yhs}_{pj} = \eta \times \sum_{t=1}^{T} \delta_{t,s,p} \times h_{t,s,j} \tag{4.33}
$$

$$
\Delta W^{hhs}_{jm} = \eta \times \sum_{t=1}^{T} \delta_{t,s,j} \times h_{t-1,s,m} \tag{4.34}
$$

$$
\Delta W^{hxs}_{ji} = \eta \times \sum_{t=1}^{T} \delta_{t,s,j} \times x_{t,s,i} \tag{4.35}
$$

With s being the topic at time t (also time $t-1$).

The intuition behind Topic-RNN is a combination of specialization and the addition of learning from a different type of context. A parallel example of why specialization works would be if one patient is suffering from a skin condition and another patient from a heart condition, then there may be better outcomes if the first patient is seen by a dermatologist and the second by a cardiologist—rather than if both are seen by the same specialist in family medicine. RNNs also only consider one type of context, which is the previous input. Along with considering the previous input, Topic-RNN also considers the topic of the word, which adds a new type of context—only the makeup of words of the topic.

To evaluate the effectiveness of prediction, we consider the case when each word (v) in the non-trained data is reflected in ϕ and the opposite case—where ϕ is the topic to word distributions from topic modeling. In the case where v is reflected in ϕ, then we simply take the output of the current s-dimension RNN (Topic-RNN-ϕ). However, when v is not already considered in ϕ, it may not always be the best decision to use the s-dimension topic model as output. In more detail: if v does not exist in ϕ_s, then this is similar to asking an untrained RNN for its prediction. In such cases it may be better to use a vanilla character RNN (Char-RNN), which contains context over every word in the corpus (not just words specific to a topic).

The difficulty in this approach is knowing when to use the Topic-RNN-ϕ and when to use the Char-RNN. Since we are predicting at the character level, we do not know the word of the current character. We do however know the previous characters. In this way we can ask each Topic-RNN-ϕ how well it can predict the next character given the existing input characters for all prefix matching words after training. With a word vocabulary of Υ, and $\overrightarrow{l}$ being the prefix $(c_1, ..., c_{t-1})$ this formalizes as:

$$A^k_{t,c_t,\overrightarrow{l}} = \overline{E^k_{t,c_t,\overrightarrow{l}}} : \forall(c_1, ..., c_t) \sqsubseteq v \in \Upsilon \tag{4.36}$$

Additionally, we can ask the Topic-RNN-ϕ how certain it is of its prediction. In cases where the Topic-RNN-ϕ is very certain then this may be reflective of a "good" guess. For certainty we measure entropy, defined as:

$$N_t = -\sum_i \hat{y}_{t,i} \times \log\left(\hat{y}_{t,i}\right) \tag{4.37}$$

We can also measure how similar the prediction is to the Char-RNN. Since the Char-RNN is a "good" baseline guess, then something that agrees with that may also be good and may be even better. We choose Jensen-Shannon distance as our similarity metric:

$$\hat{J}_t(\overrightarrow{\hat{y}'_t}, \overrightarrow{\hat{y}''_t}) = \frac{1}{2} \times \sum_i \hat{y}'_{t,i} \times \left(\ln \hat{y}'_{t,i} - \ln \frac{\hat{y}'_{t,i} + \hat{y}''_{t,i}}{2}\right) \tag{4.38}$$

$$J_t = \sqrt{\hat{J}_t(\overrightarrow{\hat{y}'_t}, \overrightarrow{\hat{y}''_t}) + \hat{J}_t(\overrightarrow{\hat{y}''_t}, \overrightarrow{\hat{y}'_t})} \tag{4.39}$$

J_t, N_t, and A_t become the features that we use to decide when to choose between Char-RNN and Topic-RNN-ϕ. But we must also consider the location of the current input. In cases where the input is the first character of a word it may always be preferable to use Char-RNN, which has more context than Topic-RNN-ϕ. Because context and location are important, we choose to feed the input features into an RNN (E-RNN) to help decide which model to use. If we assume the hidden layer size to be equal between Char-RNN, Topic-RNN-ϕ, and E-RNN, V to be the size of Υ, and ζ to be the max length word in Υ, then the memory needed would be $\mathcal{O}(K \times [U \times (C \times H + H^2) + C \times \zeta \times V])$ and the training execution becomes $\mathcal{O}(K \times [S \times (C \times H + H^2) + C \times \zeta \times V])$. In this model, the extra cost of memory and training execution time are somewhat alleviated by the prediction execution time which is $\mathcal{O}(C \times H + H^2)$.

4.2.1.2 Topic models with RNNs

Just as the addition of topic modeling features into the recurrent neural network can increase the utility of the RNN model, so too does adding the RNN to topic modeling. As noted before, the RNN has the ability to dream up sequences; that is, to create new unseen sequences with a resemblance of what the model was originally trained on. To utilize the benefits of this dreaming capability in topic modeling, the approach is to train the RNN on supervised data to help discover associations of words to topics not originally found in this supervised data. A natural fit for this approach is that of Source-LDA, which combines the existing unsupervised topics of LDA with weakly-supervised topics based off a pre-established knowledge source.

4.2.1.2.1 RNN enhanced Source-LDA

As outlined in Section 4.1.3, the theoretical generative model of Source-LDA was a natural extension to the original LDA model in that for some topics the generator goes through the original process of drawing a discrete distribution from the Dirichlet distribution parameterized by β, but for other topics the generator was assumed to be reading topics from a knowledge source and then generating a discrete distribution based off these knowledge source topics. In the RNN enhanced Source-LDA (ReSource-LDA), we assume that the generator gets so tired after spending so much time reading the topics from the knowledge source that it falls asleep a certain percentage of the time and dreams up words that belong to the current knowledge source topic.

By modeling the generative model in this way, the desideratum is to assign a higher likelihood to words that should be assigned to a certain source distribution, even though they do not show up in that source (short for knowledge source) distribution. An incomplete assumption of Source-LDA and any weakly-supervised labeling topic model is that a given source distribution for a topic contains every word that may be used when describing this topic in a corpus. This is an even looser assumption in Source-LDA where the source distributions are often based off an easy to obtain knowledge source such as Wikipedia.

The main component of ReSource-LDA is derived from the ability of the RNN to assign a higher probability to words that should belong to a knowledge source topic even though they are

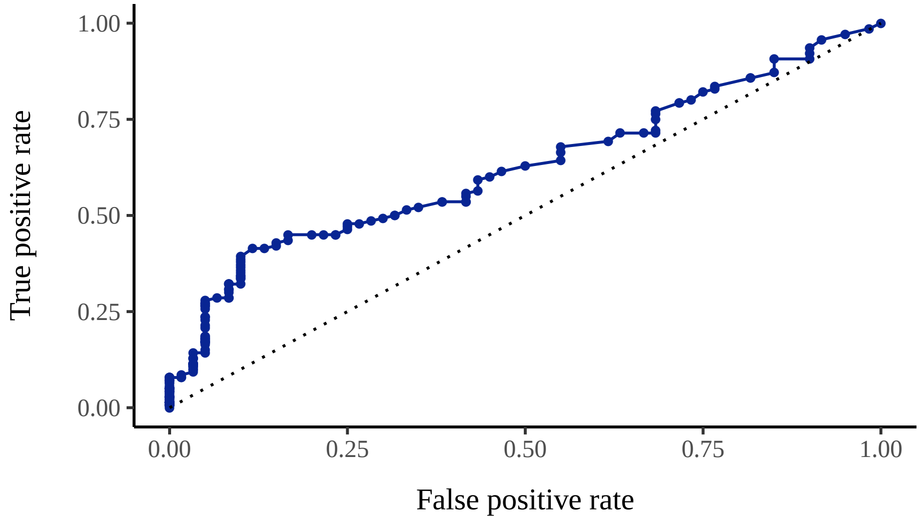


Figure 4.10: ROC plot showing the ability of the RNN to classify unseen words.

not originally contained in the topic. But what does it mean that a word should be assigned to a knowledge source topic? And can an RNN even make this prediction? We illustrate that the RNN does indeed have the potential to assign a higher probability to words that should belong to a topic via a simple case study.

Case study

A set of topics was generated by iterating through the Reuters-21578 newswire collection. For each topic, Wikipedia was queried and the resultant source distribution was constructed. This resulted in a knowledge source consisting of 80 topics and their respective articles. From this knowledge source, we train a Char-RNN on a subset of 50 randomly selected articles. After training, we compare the top 100 most common words in an article from the Reuters-21578 Wikipedia topics that do not belong in the 50 topic subset with the top 100 most common words in an article from the MedlinePlus medical collection that also do not belong in the 50 topic subset. With the objective to classify whether a word is a non-medical term, or equivalently, whether it came from the Reuters-21578 topic set. We determine the probability ($\tilde{y}$) of each word using Topic-RNN with the following equation:

$$\tilde{y}_v = \sum_t \hat{y}_{v_t} \div |v| \tag{4.40}$$

Once the probability value has been determined for each of the 200 words, we sort the values and use each probability value as a threshold for classification. The resultant ROC curve, given by Figure 4.10 shows the RNN to be useful in classification of words that should belong to the training set (are non-medical terms).

With the ability of the RNN to identify words that belong to a source topic even though they are not in the initial source topic article, it is a natural approach to integrate this feature into Source-LDA. The proposed approach is to use a metric derived from an RNN that is trained over the knowledge source to influence the model into determining which knowledge source topic a word belongs to.

The key components to building this model are when to use the RNN influence over the existing knowledge source influence, how best to derive a metric from the RNN to give weight to a word assignment, and how to apply this metric. As is shown by Figure 4.10, the results of the RNN to appropriately classify an unseen word are useful but not great. With such a low AUC, we take the approach to only use the RNN influence when no alternative is available, so we restrict the RNN to being applied only to those words which do not belong to any knowledge source topic.

	Description	**Documents**	**Words**
MedlinePlus	A consumer-friendly medical encyclopedia	961	136,000
Reuters-21578 [reu]	Manually labeled documents from the 1987 Reuters newswire	21,578	2,600,000
20-Newsgroups	Usenet articles taken from 20 different newsgroups	20,000	5,300,000
Sent-Web [KDF15]	A collection of sentiment labeled sentences	3,000	38,500

Table 4.2: Datasets used for evaluation of perplexity.

For the weight of a word to a topic, we can simply use predicted probability as given by Equation 4.40. To determine the best way to apply the metric from the RNN, we take the approach that fits simply and smoothly into the existing topic model. The approach used is to take the existing metric to use as hyperparameters to the Dirichlet distribution in the same way that Source-LDA uses the counts from the knowledge source topics. Source-LDA will then remove the unseen words for all knowledge source topics from its set of hyperparameters. This in effect creates two

	Description	Topics
MeSH [mes]	Medical subject headings	130
PhySH [phy]	Physics Subject Headings	36
ACM-2012 [acm]	ACM computing classification system	4
OAD-Wiki [oad]	Outline of academic disciplines	70

Table 4.3: Datasets used for evaluation of topic quality.

	Description	Documents	Topics
Reuters-21578	Manually labeled documents from the 1987 Reuters newswire	21,578	2,700
RE3D [re3]	A set of labeled relationship and entity extraction documents	98	2,200
Wiki-20 [MWM]	20 Computer Science papers annotated from Wikipedia articles	20	564
FAO-30 [KMK10]	Manually annotated documents from the Food and Agriculture Organization of the UN.	30	650

Table 4.4: Datasets used for evaluation of topic labeling.

distributions for each labeled topic. One which comes from the knowledge source and one from the RNN. It is important to point out that the intersection of the two vocabularies for these distributions is the empty set. This approach has the advantage of fitting smoothly into the derivations of the Gibbs sampler and does not increase the order of execution or memory requirements during inference. However, the distribution of unknown words must be built prior to inference. If D_m is the max length document in the knowledge source then the pre-inference execution time becomes $\mathcal{O}(D_m \times \zeta \times B \times [C \times H + H^2])$ and a memory requirement of $\mathcal{O}(\zeta \times [C \times H + H^2])$.

The change required to the Gibbs sampling equations are (for brevity only the affected equations are shown):

$$P(v_i|z_i{=}j,z_{\text{-}i},v_{\text{-}i}) = \int \frac{n^{v_i}_{\text{-}i,j} + (\delta_{i,j})^{g(\lambda)}}{n^{(\cdot)}_{\text{-}i,j} + \sum\limits_{l}^{V} (\delta_{l,j})^{g(\lambda)}} \mathcal{N}(\mu,\sigma) d\lambda, \; \forall v_i \in \Upsilon_k \tag{4.41}$$

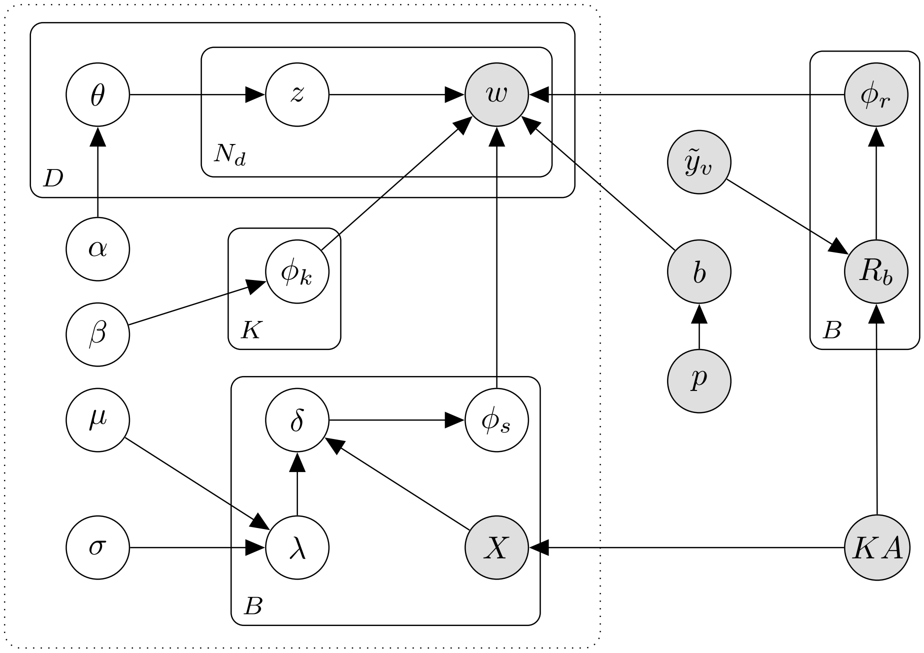


Figure 4.11: Plate notation for ReSource-LDA with Source-LDA inside the dashed box.

and

$$P(v_i|z_i{=}j{,}z_{\text{-}i}{,}v_{\text{-}i}) = \frac{n^{v_i}_{\text{-}i,j} + \tilde{y}_{v_i}}{n^{(\cdot)}_{\text{-}i,j} + 1},\ \forall v_i \in \Upsilon_r \tag{4.42}$$

where Υ_k is the vocabulary of knowledge source words, and Υ_r is the difference between the vocabulary of the corpus (Υ_c) and Υ_k.

As shown in Figure 4.11, the generative model must determine when to draw a word from the knowledge source vocabulary and when to draw from the RNN vocabulary. For this a new variable is introduced, p, which represents the probability of a knowledge source word draw from the Bernoulli distribution. In practice this p variable does not change the inference much since it is easily observed.

By modeling ReSource-LDA in this way, it is important to realize that we are intentionally not capturing all context. Since we train each RNN over the knowledge source article in a bag-of-words manner, we are loosing the connection between words. This is done because we are not guaranteed the corpus is writing about a topic the same way which it is done in the corresponding knowledge source article.

4.2.2 Evaluation

4.2.2.1 Perplexity

To test the predictive power of the various topic model based RNNs, we create an experiment to determine the ability of each RNN to correctly guess the next character given a set of input characters from a corpus.

	MedlinePlus	Reuters-21578	20-Newsgroups	Sent-Web
Topic-RNN	**0.98**	**1.99**	**2.31**	**1.5**
Topic-RNN-ϕ	**0.276**	**0.284**	**0.34**	**0.24**
CD-RNN	1.72	2.32	2.42	1.72
2-RNN	1.32	2.31	2.4	1.7
Π-RNN	8.42	8.65	8.84	8.52
Char-RNN	1.6	2.28	2.4	1.65

Table 4.5: Topic-RNN compared against baseline methods for the prediction of characters.

4.2.2.1.1 Experimental Setup

Each corpus is split into a training and test set in a ratio of 80/20. All models are then built off the training set. After 50 epochs of training, we feed characters from the test set into the models and compare the error (E) of each model. We choose 50 epochs because the error rate in the training set appears to converge, which also matches a standard used in other research [Sai11, Tam17]. To show the extensibility with more complicated cells, we repeat the experiment using only Char-RNN as a baseline for GRU and LSTM cells. We choose only Char-RNN because it is the most competitive among the baseline methods.

4.2.2.1.2 Experimental Results

The average error for each model and dataset is shown in Table 4.5. The best results come from Topic-RNN when each test word is already a part of ϕ. However, even when the test word is not known, Topic-RNN outperforms the other models. It is our contention that the vocabulary-based model is constrained by having too many outputs, while the two-hot encoding and context dependent models do not specialize. Topic-RNN can specialize while not overcomplicating the predictions and therefore leads to the best results.

The effect of input and output size seems to be substantial. Π-RNN far underperforms the others models and also has largest input and output size—while the other models have sizes more or less the same. Topic-RNN also demonstrates that specialization can improve the predictive power of RNNs, as opposed to asking a single model to learn the entire input set. The results are also promising for LSTM and GRU cells. Figure 4.12 shows the total error for all datasets when Topic-RNN and Char-RNN use LSTM and GRU cells. The results are consistent with the basic cell results and show our technique is likely applicable across different cells.

4.2.2.2 Word Prediction

Given the ability of Topic-RNN to accurately predict the next character, we aim to discover whether it can predict words as well. We compare this against a word-level RNN (Word-RNN) to highlight the effectiveness of Topic-RNN.

4.2.2.2.1 Experimental Setup

As with the previous experiment, the corpora are split 80/20 into a train and test set. Word-RNN is trained against the words and Topic-RNN against the characters and topics. After training, we determine the proportional probability of a word under the Topic-RNN model by using a simple

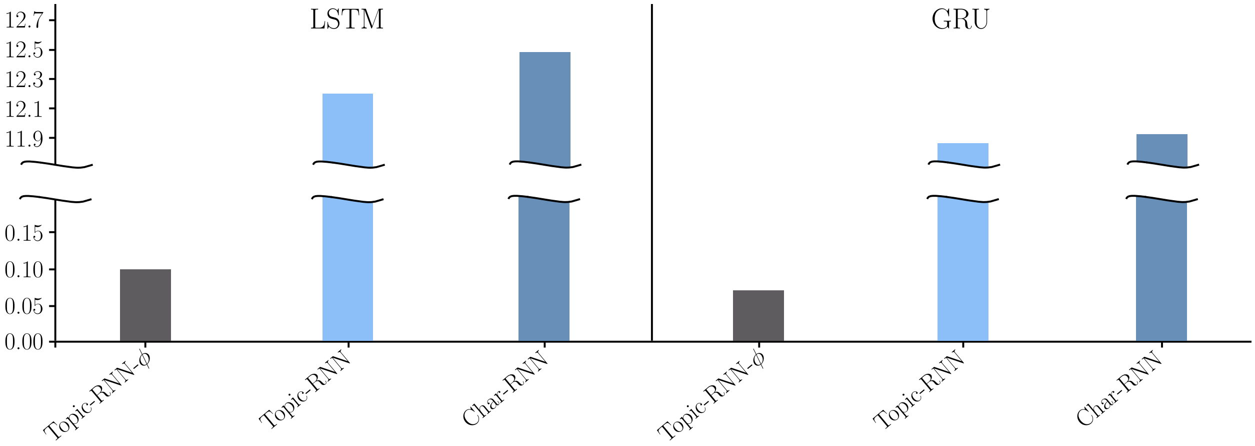


Figure 4.12: LSTM and GRU cells used in Topic-RNN compared against a vanilla Char-RNN.

average probability for each character. Given the two distributions, we can determine the error for both models.

4.2.2.2.2 Experimental Results

The results of all datasets are given by Table 4.6. Surprisingly, Topic-RNN gives better predictive error in most datasets than Word-RNN. In the other sets, such as the MedlinePlus data set, Word-RNN only does marginally better. We hypothesize Word-RNN is outperformed by Topic-RNN mainly because it is hindered by large input and output sizes; additionally, the inclusion of topics to specialize give Topic-RNN increased gains. This experiment underscores the effectiveness of adding topic modeling into RNN predictions.

	MedlinePlus	Reuters-21578	20-Newsgroups	Sent-Web
Topic-RNN	7.867	**9.97**	**10.64**	**8.09**
Topic-RNN-ϕ	**7.175**	**8.366**	**8.85**	**6.786**
Word-RNN	**7.53**	15.53	17.42	9.318

Table 4.6: Topic-RNN compared against baseline methods for the prediction of words.

	ReSource-LDA		**Source-LDA**		**EDA**		**CTM**	
	Γ	$\Lambda\%$	Γ	$\Lambda\%$	Γ	$\Lambda\%$	Γ	$\Lambda\%$
MeSH	**925.8**	**62.9**	4368.9	50.1	20390	42	2038.5	31.3
PhySH	**48.1**	**77.4**	7282.4	50.1	14449.3	41.5	984.7	40.5
ACM-2012	**3.6**	**49.9**	7637	49.9	1481.9	49.9	200.6	49.9
OAD-Wiki	**204.5**	**76**	25478	51.8	11375	37	1196.2	15.3

Table 4.7: The classification accuracy of token assignments (Λ) and perplexity values (Γ) for ReSource-LDA, Source-LDA, EDA, and CTM.

4.2.2.3 Topic Quality

As in the ability to improve the RNN model by incorporating topic modeling, the reverse process has the same capability. The Topic-RNN is added into the Source-LDA model to help in assigning meaningful labels when words appear in the corpus that do not occur in the predetermined knowledge source topics. The corpora used are detailed in Table 4.3.

4.2.2.3.1 Experimental Setup

For each hierarchical corpus, we crawl Wikipedia for the resultant source topic documents. From this set we randomly select a set of parent topics which contain at least two direct children in the found topics (we are able to determine this from the hierarchical corpus). We take two random direct children from the parent topics. We then train the Topic-RNN on only the parent topics. Next we run a modified version of the generative model for the bijective model of Source-LDA with parameters of α β, μ, σ, as $50/K$, $200/V$, 5.0, and 0.0 respectively. The modification is for each word we flip an unbiased coin to decide if we are to sample from the parent topic under the Source-LDA parameters, or from the raw child distributions. This results in a close to 50/50 split between a word coming from the parent or from the child. For each child word, we mark the topic assignment as that of the parent, and keep the parent word as assigned to the parent topic. We then run ReSource-LDA in comparison with Source-LDA, Explicit Dirichlet Allocation (EDA) [Han13], and the Concept Topic Model (CTM) [SSC11]. The goal is to determine which topic each word belongs to.

4.2.2.3.2 Experimental Results

After 1,000 iterations, we compare the perplexity and classification accuracy as a measure of goodness between the models. As is shown in Table 4.7, ReSource-LDA outperforms Source-LDA and the other models in terms of correctly assigning each word to the correct label, as well as perplexity. The baseline models are all limited in being restricted by its knowledge source distribution which leads to a low probability of a word being assigned to the topic when it is not in the knowledge source topic. ReSource-LDA rectifies this deficiency by its ability to predict words that are similar to its trained sequence, as is shown by the case study in the previous section.

4.2.2.4 Topic Labeling

Given the ability of ReSource-LDA to improve the classification of topic assignments we show how to use this improvement in identifying topics from different corpora.

4.2.2.4.1 Experimental Setup

From each corpus we take the entire set of topics associated with any article as the knowledge source. The knowledge source is constructed by taking the topic strings and querying Wikipedia for the corresponding articles. A subset of articles is randomly selected from the entire dataset to be used as the corpus. For each subset article, we also record the topics associated with that article. The topics corresponding to the article subset comprises a subset of articles out of the knowledge source. We run ReSource-LDA on the corpus with α, β, μ, σ and K set to 0.5 $(50/K)$, $200/V$, 0.7, 0.3 and 100 respectively; with V representing the size of the vocabulary. The baseline methods used consisted of Source-LDA with parameters the same as above, EDA, and the CTM. We run each model for 1,000 iterations and then compare the labeling of the inferred topics.

4.2.2.4.2 Experimental Results

After the iterations were completed, we count the number of labeled topics found that appear in the topic key of the corpus. Figure 4.13 shows the recall for each model. ReSource-LDA mostly outperforms all models finding up to roughly one third of all topics found in the key. The CTM and EDA perform poorly on this task due to their rigid nature, while Source-LDA does well because it

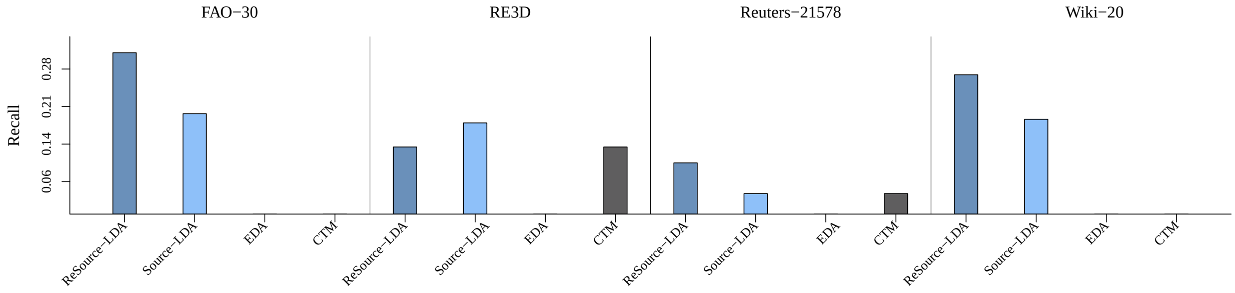


Figure 4.13: Recall demonstrating the ability of Topic-RNN over baseline methods to better retrieve the appropriate labeled topics in different corpora.

can allow for more variation. Since ReSource-LDA has the same flexibility of Source-LDA, the improvement of ReSource-LDA over Source-LDA comes from the RNN's contribution. If we take Source-LDA to be the baseline and ReSource-LDA to be the alternate variant, then the resultant p-value is 0.0551 with significance at the 0.1 level. To give the reader an idea of what it means to label a topic we select 5 different topics, their discovered label along with their top ten words and display this information in Table 4.8.

Fertilization	**Animal Diseases**	**Statistical Methods**	**Poultry**	**Disease Surveillance**
sperm	medicine	experiments	chickens	influenza
cells	vet	error	domesticated	outbreak
embryo	veterinarians	errors	fowl	pathogenic
tissue	horse	theory	domestication	infectious
reproduce	horses	bayesian	duck	epidemiology
proteins	medical	squares	goose	alert
initiate	profession	statistic	weeks	outbreaks
oocytes	london	distributions	japanese	epidemic
mating	gl	tests	chinese	emerging
dna	compilation	false	turkey	appropriately

Table 4.8: Example topics and their top 10 words found after running ReSource-LDA on the FAO-30 dataset.

4.2.3 Discussion

For the goal of improving RNNs by leveraging topic models, we show that this combination results in improved performance for the recurrent neural network. Training input from the RNN with the addition of topic modeling input can improve prediction of held out data. Although the evaluation focused on perplexity, there remains other areas worthy of investigation. We hypothesize our approach of adding topic modeling into the RNN would be useful to other tasks such as machine translation. Empirical evaluation is left as an aim in future work.

For the task of assigning meaningful labels to topics, the RNN can improve upon the performance of existing methods, especially when there exist a large portion of words not found in the weakly-supervised input set. An open area of interest would the addition of existing work which focuses on the non-labeled topics [DWG17, Ngu15] to see if these changes propagate to the labeled topics as well. We validate our approach of aiding topic labeling experimentally as presented in this paper.

This work is an investigation into the utility of combining RNN and topic models. We show that this combination results in improved performance for both models. Training input from the RNN with the addition of topic modeling input can improve prediction of held out data. Likewise, for the task of assigning meaningful labels to topics, the RNN can improve upon the performance of existing methods, especially when there exist a large portion of words not found in the weakly-supervised input set. Topic-RNN represents a novel and outperforming method of combining topics with the recurrent neural network. Our analysis shows the advantage of learning from both previous data as well as data partitioned by topic. Topic-RNN's improvements over existing topic-based methods have been validated experimentally. Under these experiments, and likely others the combination of topic models and RNNs leads to improvement in both models.

4.3 KnowledgeRank

Another deficiency of Source-LDA is the increase of execution time proportional to the size of the knowledge source. Indeed, when the input approaches the order of 10^3 knowledge source topics the model becomes unfeasible to run. We approach this problem from the angle of pre-inference filtering. The technique we develop utilizes rankings to eliminate knowledge sources a priori. We also discover this ranking can improve the model much like ReSource-LDA.

Although lately topic modeling research seems to be directed towards neural topic modeling (NTM) [Dua21, Che21a, Rez20], traditional, Bayesian based topic models (BTM) offer a viable alternative to deep learning approaches. Bayes approaches may be preferable when (1) using commodity or legacy hardware, as the NTM often requires a more complex setups (such as utilizing a GPU), (2) a document-to-topic (θ) distribution is needed, since for the NTM, θ is often associated with a batch parameter and reused for multiple documents [Dua21, Che21a, Rez20], and (3) for more interpretable topics [DB21, CBG09] since the high perplexity of the NTM may lead to lower interpretability [CBG09], and the recent work challenges the goodness of traditional pointwise mutual information (PMI) based interpretability scoring often reported in NTM results [DB21, Geo21]. The latter scoring method [Geo21] may be the preferred approach to take for estimating interpretability of topic models, however we take direct human based scoring to be a stronger approach to evaluate interpretability.

As demonstrated in Figure 4.15 (with the variables described in Table 4.9 and Table 4.10), the traditional probabilistic topic model outputs a distribution of numeric topics for each document and a distribution of words for each numeric topic [BNJ03]. These latter distributions comprise the "topics" in topic modeling. As such, a "topic" is just a distribution over words with a numeric label. However, the numeric label fails to summarize the distribution semantically. As we can see from Experiment 4.1.4.3, semantically labeling each topic gives the end user a quick understanding of what each topic represents, improving the interpretability. These labels can also be used in downstream processes such as graph-based summarization systems [Ble12, AOC16], consensus building [LGY20] and scene identification [ZGX21]. However, assigning an accurate label to a topic is no trivial task.

To assign semantic labels to topics, one can run an unsupervised topic model and then choose labels after inference [LGN11, MSZ07, MMZ12, SXW15, MCN13, HHK13, Pec10]. However, this can lead to problems with the topics themselves as the clusters tend to combine two or more semantically different topics [WTW17]. An example of this given by Case Study 4.0.0.1.

As previously discussed, a second approach to semantic topic labeling involves using a supervised input set and has shown the ability to label the topic as necessary [JIU12, BM07, LSJ08, RHN09]. This approach requires many labeled input that may be time-consuming or expensive to acquire. A compromise between automatically assigning labels while requiring little effort to obtain a labeled input is given by Source-LDA in in Section 4.1.3. Source-LDA is part of a larger class of models, referred to as weakly-supervised topic models. To allow for a labeled input set that is easier to obtain, weakly-supervised topic models [WTW17, Han13, SSC11, SGP20, GKM21] use existing knowledge sources as the weakly-supervised input to label topics. The knowledge sources consist of articles turned into distributions and can be transformed into *knowledge source topics* ($\hat{\phi}$). A formal definition of *knowledge source knowledge source topics*, and *weakly-supervised topic modeling* is given in Section 2.2.6. To further illustrate the concepts of weakly-supervised topic modeling, consider the following simple example.

4.3.0.1 Wikipedia Case Study

At the time of this writing, if we open a web browser and go to Wikipedia[2] and search for "grape," the returned article ($\hat{A}$) would start with the following text:

A grape is a fruit, botanically...

If we take the above to be the full article, then the knowledge source topic ($\hat{X}$) for "grape" can be formed by taking a count of each word ($\hat{w}$) in the article and dividing each word by the total number of words. For the "grape" example, the knowledge source topic is the probability vector $[\frac{2}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}]$ with the index of the probability vector mapped to the word vector [a, grape, is, fruit, botanically].

If we continue the above for a set of articles from Wikipedia, the set of articles becomes the *knowledge source* (KS). We follow the above procedure from the knowledge source to get a set of knowledge source topics. These knowledge source topics are then used in the corpus's theoretical generative model. Before generating the corpus, we determine the total number of topics (K) and vocabulary size (V). For each *topic*, we sample from a Dirichlet distribution that may or may not be influenced by an individual knowledge source topic. If a knowledge source topic influences the topic, the *topic label* becomes the article's title from which the knowledge source topic was created ($\hat{L}$). Each *document* in the corpus is generated by first sampling a *topic* from a discrete distribution of size K. After the topic is sampled, a *word* is chosen by sampling from the topic's discrete distribution (ϕ) of size V. During inference, the topic model takes as input a set of knowledge source topics that may or may not be used in the final output of topics. Because the output is dependent on a subset of labeled data, we refer to this type of topic modeling as *weakly-supervised topic modeling*.

One drawback of weakly-supervised topic modeling is the excess knowledge source topics used as input. Since there is a more relaxed constraint of not needing to know precisely which knowledge source topics are relevant to a corpus, there tend to be many knowledge source topics ultimately discarded. Existing approaches used to determine which topic to discard are based on counting or some form of clustering. However, counting is problematic because it is too simple and often discards *important* knowledge source topics due to not having a high count. In this context

[2] https://en.wikipedia.org/w/index.php?title=Grape&oldid=908871054

we take important topics to be topics which are used in the generative model of the corpus. Even worse is clustering, which only considers some distance metric between the topics and does not consider how many assignments of words have been made to the topic. We illustrate these concepts in another simple case study.

4.3.0.2 EHR Case Study

We are given the task of labeling patient notes from a small set of electronic health records. Given that we know we are in the medical domain, we suppose all possible and relevant topics for any patient note to be in the following set:

$\mathbf{\hat{A}_1}$ - Cancer, cancer, tumor, chemotherapy
$\mathbf{\hat{A}_2}$ - Heart attack, heart, attack chest
$\mathbf{\hat{A}_3}$ - Dementia, brain, memory, dementia
$\mathbf{\hat{A}_4}$ - Diabetes, blood, sugar, insulin

Next, we wish to obtain topics and corresponding labels for a corpus of two documents d_1 and d_2, given as:

$\mathbf{d_1}$ - cancer, chest, attack
$\mathbf{d_2}$ - tumor, heart, chemotherapy

A good weakly-supervised topic model would start by considering the entire knowledge source of ($\mathbf{\hat{A}_1}$,$\mathbf{\hat{A}_2}$,$\mathbf{\hat{A}_3}$,$\mathbf{\hat{A}_4}$) but would eventually end up with document-token to topic assignments of:

$\mathbf{d_1}$ - cancer^1, chest^2, attack^2
$\mathbf{d_2}$ - tumor^1, heart^2, chemotherapy^1

With topic 1 (after the topic model interference is complete) mapped to $\mathbf{\hat{A}_1}$ and topic 2 mapped to $\mathbf{\hat{A}_2}$. Since $\mathbf{\hat{A}_1}$ and $\mathbf{\hat{A}_2}$ are referenced in the final document-token assignment, we consider these *relevant* or *important* topics. Additionally, since $\mathbf{\hat{A}_3}$ and $\mathbf{\hat{A}_4}$ were not referenced by any document-token assignment to topic, we delegate these to be *discarded* topics.

It is essential for the weakly-supervised topic model to determine which topics are relevant and which topics to discard. What is needed is some way to rank the topics by order of importance to a corpus. A better ranking of topics can select the relevant topics and discard the less important ones. Counting can be used for ranking, but this leads to the problems discussed previously. One method for ranking which has already shown promising results is PageRank [PBM99]. PageRank finds the importance of a node by considering the importance of the connecting neighbors in a recursive fashion. This approach helps determine the importance of websites in the world wide web.

With the success of PageRank in the world wide web, it is a natural approach to apply the techniques of PageRank to other ranking problems, such as the ranking of article-topics. The main obstacle of using PageRank for knowledge source rankings is representing the knowledge source as a graph consisting of nodes and edges. In most cases, a knowledge source consists of a collection of articles, i.e., Wikipedia articles corresponding to MedlinePlus[3] headings. However, there are knowledge sources that already take the form of a graph, such as the Unified Medical Language System (UMLS)[4]. Ontologies and other compendia exist that take the form of entities as nodes and relationships among entities as edges. For these cases we still need to determine a way to effectively rank the nodes and edges which is applicable in the context of weakly-supervised topic modeling.

Still, with the desiderata to increase applicability, we must consider how to rank existing article-based knowledge sources. This work presents a novel way to aid topic models that already have a knowledge source associated with the corpus (weakly-supervised topic models). Our technique applies to both graph-based and article-based knowledge sources. When we have both a graph and article-based knowledge source, we can take the topic labels from the article headings and emphasize these nodes in the graph-based knowledge source. When comparing the results after ranking, we can select the subset of nodes corresponding to article labels. We also formulate similar approaches for article-only and graph-only knowledge sources.

Ranking the article-topics allows weakly-supervised topics models to take any input knowledge source regardless of size. Currently, as validated in Experiment 4.1.4.5, for our weakly-supervised topic model, a knowledge source input size of just 1,000 article-topics results in inference iteration times that are too high to be practical. Our solution is to rank the article-topics using our ranking method pre-inference and filter out low scoring article-topics. We can then input the filtered knowledge source into the weakly-supervised topic model and proceed as usual.

[3] https://www.nlm.nih.gov/medlineplus/

[4] https://www.nlm.nih.gov/research/umls/

Knowledge source rankings are not only limited to preprocess filtering. The rankings are also applicable during topic modeling inference to help existing weakly-supervised algorithms determine which topics should be removed. We can also use knowledge source rankings in a stand-alone topic model or in the generative model alongside existing weakly-supervised topic models.

The intuition behind our ranking method is like that of TextRank [MT04]. This established method ranks sentences in a document to determine a sentence used to summarize the document. Similarly, and with some modifications, we should be able to develop a technique to determine a ranking of article-topics relevant to a corpus. Additionally, knowledge source preprocess filtering has already been shown to improve text-related tasks [KDA20], while utilizing outside text-based information and graph representations are shown to yield improved results as well [AG18, OEA20, AMC19].

4.3.1 Motivating Examples

We provide a few small examples to help understand the intuition behind using ranking algorithms for weakly-supervised topic models.

4.3.1.1 Graph-based knowledge sources

The proposed ranking algorithm allows for the inclusion of graph-based knowledge sources into the weakly-supervised topic modeling process. We can see from Section 4.1, Source-LDA only allows for article-based knowledge sources. This same limitation is intrinsic to other weakly-supervised models as well [Han13, SSC11]. The addition of an extra source of outside knowledge should serve to enhance existing weakly-supervised methods. For example, suppose we are working with a corpus of PubMed[5] articles, and we observe the word *acetylsalicylic acid* (commonly known as aspirin). We are now trying to classify this word as belonging to either *Cerebral infarction* or *Alzheimer's disease* using an article-based knowledge source derived from Wikipedia. However, neither the article for *Alzheimer's disease* nor *Cerebral infarction* contains the word *acetylsalicylic acid*[67] (as well as aspirin), leaving the model to choose the topic assignment from outside the knowledge

[5] https://www.ncbi.nlm.nih.gov/pubmed/

[6] https://en.wikipedia.org/w/index.php?title=Cerebral_infarction

[7] https://en.wikipedia.org/w/index.php?title=Alzheimer%27s_disease

source. However, suppose we were to leverage the graph-based knowledge source UMLS. In that case, we have a direct connection[8] between *Cerebral infarction* and *acetylsalicylic acid*—whereas none exists between *acetylsalicylic acid* and *Alzheimer's disease*. This extra information can help to classify *acetylsalicylic acid* to *Cerebral infarction* over *Alzheimer's disease* at a more accurate percentage[9]

4.3.1.2 Overlapping topics

Another advantage of weakly-supervised topic ranking comes from leveraging information from overlapping topics. In this example, suppose we try to classify w_1 as belonging to t_1 or t_2, and w_1 is not in either t_1 or t_2's knowledge source article. However, a third topic, t_3, contains w_1 and w_2, which t_1 shares. Furthermore, t_2 does not share any words with t_3. Thus, ranking can help prefer t_1 over t_2 as the score is propagated based on distance. However, other methods: counting, Gibbs-based, etc., cannot give such an advantage.

4.3.1.3 Discarding topics

At some point, the topic model must choose to discard topics assumed not to be used in the generative model. Existing methods use counting, assuming that if a topic was used in the generative model, then there will be more word assignments to that topic than a topic not used in the generative model. However, this may not be the best way to eliminate topics. Consider the example shown in Figure 4.14. Here we have modeled assignments of words to topics as a graph with a word having an outgoing edge to a topic if that word is assigned to that topic. If we must discard one topic out of the existing topic set, counting would choose t_4. However, a better topic to discard would be t_1 since the words assigned to t_1 are shared words that could easily be from other topics. Ranking would consider the context of the words assigned to t_4 to choose to keep t_4 over t_1.

[8] C0007785 RO/may_be_prevented_by C0004057

[9] Based on a PubMed MeSH term search of *acetylsalicylic acid* and *cerebral infarction* versus *acetylsalicylic acid* and *alzheimers disease*, yielding 488 to 58 results respectively.

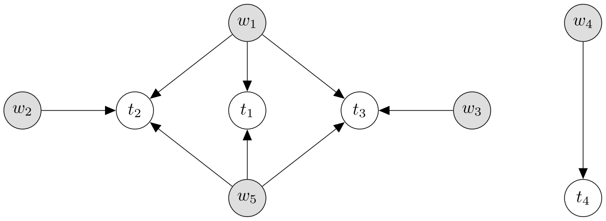


Figure 4.14: An example graph representation of word and topic assignments.

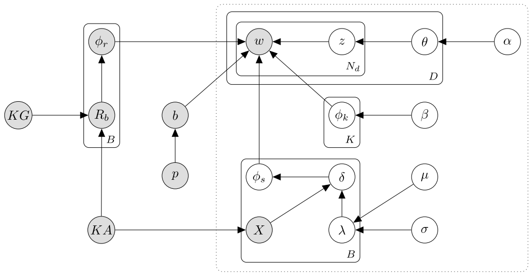


Figure 4.15: Plate notation for Rank-LDA with the dashed box representing Source-LDA. All variables are described in Table 4.9 and Table 4.10

4.3.1.4 Research objectives

Section 4.1.4 shows us that weakly-supervised topic models, in particular Source-LDA, have the ability to improve upon traditional topic modeling in two ways: (1) an increase in interpretability and (2) the labeling of topics. So then why hasn't weakly-supervised topic models been widely adopted as a standard for all Bayesian topic modeling? By design, the weakly-supervised input is effortless to obtain for most corpora. With a model that is easily adaptable to all Bayesian topic models, why not have more interpretable topics that are labeled? It seems that in most applications of topic modeling, this could only help. One reason may be due to the high execution time. At large weakly-supervised input sizes, the model running times become unfeasible. It is our objective to resolve this inadequacy. By removing the burden of high execution times while still maintaining the benefit of weakly-supervised models, we hope this topic modeling technique takes a step toward being the approach used in all topic modeling. Additionally, we seek to use the same techniques to speed up the execution of weakly-supervised topic models to further improve the interpretability and perplexity of these models. The desiderata of this work is to give existing consumers of weakly-supervised topic models another tool that can improve execution time, perplexity and interpretability—and to convince any topic modeler that weakly-supervised topic models are an effective enhancement to existing topic models on just about every dataset.

Symbol	Description
w	A word in a document of size N_d
z	The topic corresponding to w
θ	A distribution over topics for each $d_i \in D$ documents, parameterized by α
α	The Dirichlet hyperparameters for each K topics
N_d	The number of words in $d_i \in D$ documents
D	The number of documents in the corpus
ϕ_k	A distribution over words for each $k \in K$ topics, parameterized by β
β	The Dirichlet hyperparameters for each w words
B	The number of knowledge source topics
K	The number of latent topics
ϕ_s	A distribution over words for each $b \in B$ topics, parameterized by δ
δ	The Dirichlet hyperparameters for each word in $b \in B$ topics. The value is a result of a function applied to X and λ
μ	The mean to the normal distribution
KA	An article-based knowledge source
X	The count of each word in $a \in KA$ knowledge source article
λ	A latent number that signifies how far ϕ_s deviates from the corresponding frequency distribution
σ	The standard deviation to the normal distribution

Table 4.9: Notations used in Source-LDA.

4.3.2 Methods

With the desideratum to leverage graph-based knowledge sources in topic modeling, we must first model the weakly-supervised input in a way that maximizes the effectiveness of the ranking. We introduce our method, KnowledgeRank, for constructing a graph-based representation of a knowledge source for ranking the appropriate nodes and edges.

4.3.2.1 Graph-based knowledge sources

In cases where the only weakly-supervised input set is already in the form of a graph, we can simply use the given structure as the model for KnowledgeRank. However, what is not entirely clear is how to obtain the labels. Many ontologies or other compendia consist of concept nodes that can be

Symbol	Description
ϕ_r	A distribution over words for each $b \in B$ topics, parameterized by R_b
KG	A graph-based knowledge source
R_b	The Dirichlet hyperparameters for each word in $b \in B$ topics influenced by KnowledgeRank
p	Bernoulli distribution parameter
b	Draw from the Bernoulli distribution parameterized by p to determine which knowledge source w is drawn from

Table 4.10: Notations used in Rank-LDA.

Symbol	Description
$R_g(n)$	The rank score for a node n in graph g
$I(m)$	The set of all nodes with incoming edges into node m
$O(n)$	The set of all nodes with incoming edges originating from node n
C_n	The count of word n in a corpus
$\hat{D}$	A corpus
d	Damping factor
ζ_g	An input parameter over the interval [0,1] specifying the ranking importance of frequent words in a corpus with corresponding knowledge source graph g
N	The set of all nodes
$P(w_i \vert t)$	The probability of word w_i given topic t
$S(t,w_i)$	The set of nodes in the shortest path from node t to node w_i
$X(m,n)$	The number of times word n appears in topic m's knowledge source article
z_i	The ith topic assignment
$\overrightarrow{z}_{-i}$	A vector of all topic assignments minus the ith assignment
b	A variable representing the draw from a Bernoulli distribution
$n^{w_i}_{\text{-}i,j}$	The number of assignments of word i to topic j minus the current assignment
$n^{(\cdot)}_{\text{-}i,j}$	The number of assignments to topic j minus the current assignment

Table 4.11: Explanations for variables used to describe the methodology behind KnowledgeRank.

used as labels for topics, however there often exists noisy word nodes that would be inappropriate labels for a given topic (and no way to differentiate between a concept node and word node). For example, in the NIF ontology, a given node may correspond to the word "of," which obviously would not be a good label for any topic. These less applicable words exist because the ontologies or other compendia are often built from domain specific free-text. This curated data source can still be helpful for topic models, but we must first find the appropriate labels.

Graph-based ranking models have already established the ability to find the most important word in a sentence [MT04]. It follows that similar techniques can find the most important node from a set of nodes. If we apply the ranking algorithm to a knowledge source graph, we can determine the labeling for a topic based on the highest scoring nodes.

By applying the ranking in this way, we can obtain the most important nodes in the graph, but in some cases, we may want to let the corpus give us insight into the importance of a node. It is plausible that a word used more frequently in a corpus should be considered more important in the representative graph than one that is used very seldom. In other cases, this weighting is not so important. To account for these cases, we can augment the original PageRank formula to consider these weights and the associated importance of the weighting (ζ_g) as:

$$w_1 = \sum_{n \in I(m)} \frac{R_g(n)}{|O(n)|} \tag{4.43}$$

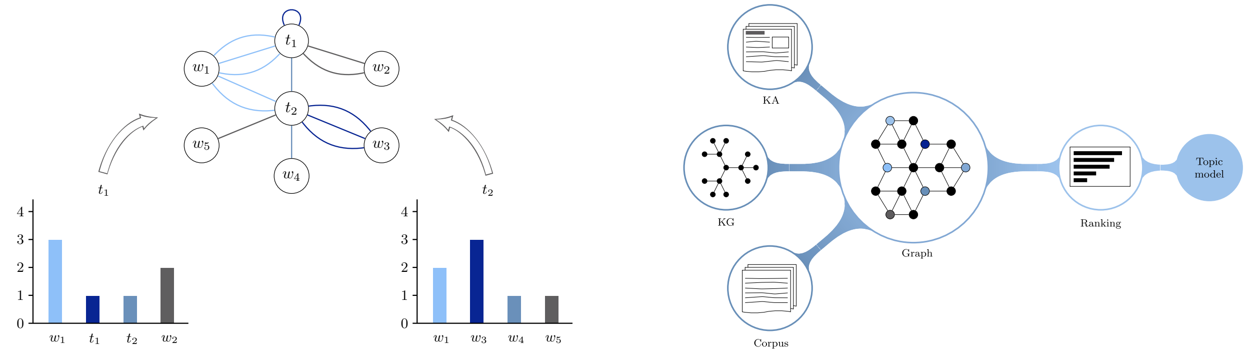
$$w_2 = \sum_{n \in I(m)} \frac{C_n}{|D|} \cdot R_g(n) \tag{4.44}$$

$$R_g(m) = \frac{1-d}{|N|} + d \cdot [\zeta_g + (1-\zeta_g) \times w_2] \cdot w_1 \tag{4.45}$$

C_i is the count of word i in the corpus D, and ζ_g is defined over the interval 0 to 1.

We can also use this information in the generative model itself. Given that we only have the graph-based knowledge source, we can construct a distribution over the vocabulary using Equation 4.45. In more detail: we can form a distribution over the vocabulary by starting at a topic label node, t, and normalizing the probability of arriving at each word in the vocabulary. The distribution can be calculated by considering the path a random surfer takes to each node with the restriction that the random surfer starts at each labeled node. This function is given as:

$$P_g(w_i|t) \propto \prod_{m \in S(t,w_i)} R_g(m) \tag{4.46}$$



(a)

(b)

Figure 4.16: A graph-based representation of two topic (t_1, t_2) histograms corresponding to knowledge source articles (a) alongside a diagram representation of the pipeline needed for pre-inference filtering (b).

The advantage of this approach is that the change required to infer the model's hidden variables can easily be adapted to any weakly-supervised topic model's Gibbs sampling equation. We can precompute the probabilities and then use the distributions the same way as a word distribution from an article-based knowledge source. In this approach, we add curated outside knowledge while still allowing LDA to cluster the topics.

4.3.2.2 Article-based knowledge sources

For those knowledge sources consisting only of articles, we can model the articles into a graph and then run our ranking algorithm. Our approach connects each topic node to each corresponding source article word. Because frequent words in an article are assumed to be more important to topic identification, we would like to give these words more weight in our graph representation. We add this weighting by creating an edge (from topic to word) for each token in an article. For example, take the two histograms corresponding to a knowledge source article (article-topic) shown in Figure 4.16(a). In this example, each t_i represents a knowledge source topic label (or article heading) with each w_i as a non-topic label word in knowledge source topic i. We model the edges as undirected, resulting in $I(n) = O(n)$. Note that an article-topic can have in its article a word that is also a label for another article-topic. Also note that a word can be a non-topic label word (shows up in the body of the text) and the knowledge source topic label (the article heading) in the same knowledge source topic (such as t_i). The change required to the ranking algorithm is the weighting of each node. This change gives us:

$$w_3 = \sum_{n \in I(m)} \frac{R_a(n)}{|O(n)|} \tag{4.47}$$

$$w_4 = \sum_{n \in I(m)} \frac{X(m,n)}{\sum\limits_{m \in O(n)} X(m,n)} \cdot R_a(n) \tag{4.48}$$

$$R_a(m) = \frac{1-d}{|N|} + d \cdot [\zeta_a \times w_3 + (1-\zeta_a) \times w_4] \tag{4.49}$$

where ζ_a, a parameter defined between 0 and 1, lets us specify the importance of weighting the edges over a PageRank score, and $X(m,n)$ is the count of the number of token assignments word n has in knowledge source topic m.

We can then use the graph-based representation in tasks mentioned in the graph-based knowledge sources section with this representation. This method would be beneficial in preprocessing to decrease some of the non-important topics.

4.3.2.3 Graph and article knowledge sources

Having both graph and article-based knowledge sources brings a more extensive set of information into the topic model and thus can lead to better labeling of found topics. Given that we already have the graph form, we can apply the ranking algorithm to preprocess the existing knowledge source articles. We would want to let the corpus tell us about the importance of a word, but we also want to consider how important it is in the knowledge source article. For this, we make a change to the ranking calculation that allows for this weighting:

$$w_5 = \zeta_g \times w_1 + (1-\zeta_g) \times w_2 \tag{4.50}$$

$$w_6 = \zeta_a \times w_3 + (1-\zeta_a) \times w_4 \tag{4.51}$$

$$R(m) = \frac{1-d}{|N|} + d \cdot w_5 \cdot w_6 \tag{4.52}$$

We can use this ranking to perform all the tasks previously mentioned, such as pre-inference topic filtering, as we diagram in Figure 4.16(b). Additionally, this ranking can be helpful in the inference stage of existing weakly-supervised topic models. During inference, the topic model must decide which topics to keep and which ones to discard. To determine which topic to discard, the algorithm considers a simple observable property such as the count of assignments to a topic. This decision can lead to problems such as when two related topics are used in a corpus, and thus one takes most of the overlapping word assignments. The topic with the smaller number of overlapping word assignments is then discarded. When using clustering algorithms, the same problem exists, limiting the similarity of two topics to a distance measure. Compared to clustering, using counts has more of an underlying intuition. We can use the ranking methods described previously as a third way of determining which topics to discard. After obtaining a ranking, we can simply remove an appropriate number of low-scoring topics.

Both knowledge sources can also be combined in a topic model that leverages the graph-based connections to increase the probability of words being assigned to the appropriate source topic when they do not appear in the knowledge source article. An incomplete assumption of article-based knowledge sources is that they contain every word for which the generative model would use to write about a particular topic, but this is certainly not the case. It is entirely possible that important words about a topic may not show up in a random document describing that topic. Graph-based knowledge sources can help add more information into the model. The generative process can be changed to allow for this synthesis of information. The change required to Equation 4.46 is:

$$P(w_i|t) \propto \prod_{m \in S(t,w_i)} R(m) \tag{4.53}$$

The graph-based knowledge source exerts an influence over the word assignments differently than that of an article-based distribution—due to each knowledge source containing independent data. To handle both of these influences, we can place a Dirichlet prior over the selection between the models. Based on an input hyperparameter, the data will decide which distribution to select—and then we sample from this multinomial to determine which knowledge source is used to select the word. A more straightforward approach assumes that the vocabulary of the samples for the different types of knowledge sources is disjoint. This approach allows the generative model to sample the knowledge source choice variable (b) from the Bernoulli distribution, parameterized by p. During

	Description	**KA**	**KG**	**D**	**K**
MeSH	Medical subject headings	Wikipedia	UMLS	2,000	56,326
CiteULike-180	Manually tagged scholarly papers	Wikipedia	WordNet	182	1,660
FAO-30	Manually annotated documents from the Food and Agriculture Organization of the UN.	Wikipedia	WordNet	30	650
SemEval-2010	Scientific articles with manually assigned keyphrases	Wikipedia	WordNet	244	3,107
Reuters-21578	Manually labeled documents from the 1987 Reuters newswire	Wikipedia	WordNet	21,578	2,663

Table 4.12: Non-hierarchical datasets used for evaluation of KnowledgeRank.

inference, p is easily observed and does not factor into the inference other than to determine which calculation to use.

As shown in Figure 4.15, we can build a Gibbs sampler from the generative model. The choice variable, b, should be included in the Gibbs sampling and used to determine which distribution to sample from. The step sampling for $b = 0$ is the same as Equation 4.9. For $b = 1$, the step sampling is drawn from the proportional probability of [$P(z_i{=}j|\overrightarrow{z}_{\text{-}i})$ unchanged and omitted]:

$$P(z_i{=}j|\overrightarrow{z}_{\text{-}i},w_i,b{=}1) \propto \frac{n^{w_i}_{\text{-}i,j} + P(w_i|j)}{n^{(\cdot)}_{\text{-}i,j} + 1} \tag{4.54}$$

We take this approach to be Rank-LDA.

Rank-LDA is shown in Figure 4.15 as an extension to Source-LDA however a similar extension to any weakly-supervised topic model would result in a congruent construction. Rank-LDA uses the article-based knowledge source (KA) in two ways. The first being the original way used in the weakly-supervised topic model. The second is to provide supplemental support to the graph provided by KG. The intuition is that both KA and KG provide partial information about a topic and that combining them can only help. Additionally, by turning KA into a graph, we take advantage of ranking over counting, which gives us the advantages discussed in the motivating examples section. One disadvantage of this approach is that it does not consider the quality of the knowledge sources (KA and KG), thus weighting them equally. A poor-quality knowledge source could add noise leading to less desirable results [NND17]. Knowledge source weighting and optimization are left as an open research area.

	Execution time		**Preprocessing**			**Trade-off**
	$\overline{R}^2_{f(x)}$	$\overline{R}^2_{f(x^2)}$	AUC_{rank}	AUC_{vote}	AUC_{js}	r
MeSH	**0.905**	0.768	**0.418**	0.264	0.02	-0.963
CiteULike-180	**0.519**	0.299	**0.244**	0.192	0.192	-0.529
FAO-30	**0.344**	0.186	**0.269**	0.238	0.238	-0.836
SemEval-2010	**0.493**	0.28	**0.229**	0.104	0.104	-0.821
Reuters-21578	**0.437**	0.236	**0.223**	0.164	0.164	-0.764

Table 4.13: Metrics describing the execution of KnowledgeRank in the preprocessing stage.

4.3.3 Results

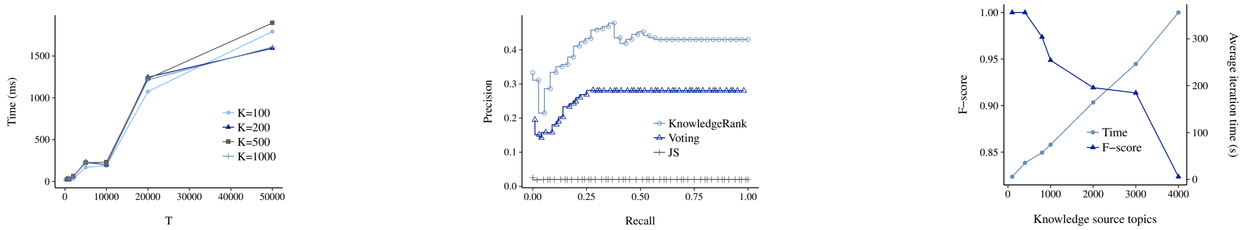
Knowledge source rankings are applied in various experiments to show the utility of KnowledgeRank.

4.3.3.1 Datasets

To examine how well our algorithm performs across different datasets, we collected datasets across various domains and varying sizes. Details and metrics are provided in Table 4.12 and Table 4.3. The datasets can be partitioned into two sets: hierarchical and non-hierarchical. For the non-hierarchical datasets, we required a corpus with topics labeled by a human annotator. Each dataset was taken from previous work on similar topic modeling tasks [MFW09, KMK10]. The datasets were preprocessed differently depending on the experiment. More details are provided in each experiment's experimental setup. The hierarchical datasets consist of parent, child relationship topic pairs. Each child was restricted to one parent, while each parent could have multiple children. Thus the network structure resembled a forest as opposed to a graph. More details about construction are given in the experimental setup for the hierarchal experiments (Section 4.3.3.6).

4.3.3.2 Execution Time

For KnowledgeRank to be helpful in preprocessing, we seek to add a filtering approach that does not significantly add to the overall time needed to perform topic modeling. An execution cost that is minuscule compared to the time needed to complete Gibbs sampling of a corpus is ideal, given that execution times of weakly-supervised topic models can be quite expensive—as an example, take Experiment 4.1.4.5.



(a) (b) (c)

Figure 4.17: Results showing the execution time for running KnowledgeRank (a), the precision-recall curve for selecting the topics used in the generation of the corpus (b), and the trade-off between execution time and F-score (c). All results shown are in the preprocessing stage for the MeSH dataset.

We run KnowledgeRank as the preprocessing step on a dataset that consists of articles from Wikipedia corresponding to MeSH terms. We seek to obtain the best K topics from a superset of T knowledge source distributions. With K taken as 100, 200, 500, and 1,000 topics. T also varies from 0 to 50,000 superset topics. Figure 4.17(a) shows that the execution time increases linearly with an increase of T. The different values of K do not significantly impact the results, and even at extreme values of K and T, the total execution time is relatively small—at 1.5 seconds, this is much less than the time taken in Experiment 4.1.4.5.

The same experiment was performed on each of the non-hierarchical datasets. To show the linearity of the execution times, we compare the average coefficient of determination of a linear function fit to the data against a quadratic function. The functions were fit using the least squares approach. As shown in Table 4.13, the results show more of a linear relationship than a quadratic relationship for the execution times of KnowledgeRank in preprocessing.

4.3.3.3 Preprocessing

A proposed advantage of KnowledgeRank is the ability to appropriately determine which topics are used in the generation of a corpus. We show the utility of KnowledgeRank in this task by comparing it to baseline methods. We consider only baseline methods that require much less computation cost than that of topic modeling.

4.3.3.3.1 Experimental Setup

We generate a corpus by first taking a random subset of MeSH article headings and then combine them with all MedlinePlus article headings. For each article heading, we search Wikipedia for the corresponding article. If a query leads to no results or multiple results, we discard the article heading. The process results in 4,300 found Wikipedia articles. Each Wikipedia article is then turned into a histogram over the set of words in the article. Given the histograms corresponding to Wikipedia articles, we generate a corpus of 2,000 documents, each consisting of an average of 500 words using the Source-LDA generative model. The Source-LDA parameters are K, α, μ, and σ set to 100, 0.5, 5, and 2, respectively. For KnowledgeRank, we take as input the SNOMED CT[10] subset of the UMLS. The graph is filtered by removing any node whose corresponding string label does not occur in the corpus. We then run KnowledgeRank on the filtered graph. The first baseline method is based on voting, where one vote is cast to each topic for every word in both the corpus and the corresponding knowledge source article. A second baseline method is constructed by taking each document as a discrete distribution and scoring the likelihood of a topic existing in the corpus by comparing the Jensen-Shannon (JS) divergence. We then repeat this experiment for all datasets and record the area under the curve (AUC) of the precision-recall curve.

4.3.3.3.2 Experimental Results

The corpus and knowledge sources are used to determine the ground truth of topics used in the corpus. Figure 4.17(b) shows the precision-recall curve for each model's ability to determine whether a topic was used in the corpus. KnowledgeRank outperforms the baseline methods significantly as the JS divergence baseline method has a hard time separating the mixtures, and voting is not refined enough to accurately capture the matching. Bringing into the model the outside information of the UMLS allows for a more accurate determination of correct topics, while doing so in a computationally efficient manner. Table 4.13 confirms that KnowledgeRank is consistently better in preprocess filtering than baseline methods.

[10] http://www.snomed.org/

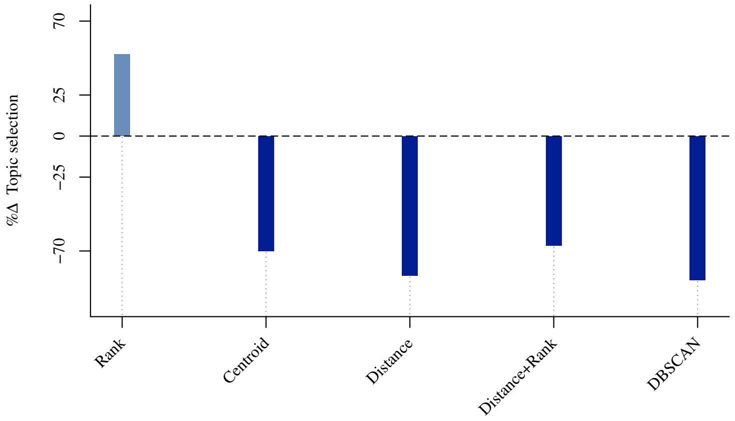


Figure 4.18: A bar chart representing the increase in topic filtering decisions made during inference using KnowledgeRank and clustering-based methods as a percentage over the naive approach of simple counting of assignments to each topic using the MeSH dataset.

	%Δ **Topic selection**				
	Rank	Centroid	Distance	Distance+Rank	DBSCAN
MeSH	**50**	-70	-85	-66.667	-87.755
CiteULike-180	**10**	0	0	0	0
FAO-30	**6.25**	0	0	0	6.25
SemEval-2010	**30**	0	0	0	0
Reuters-21578	**50**	0	0	0	0

Table 4.14: The increase in topic selections for all datasets when using KnowledgeRank and clustering-based methods over simple counting during topic modeling inference.

4.3.3.4 Preprocessing Trade-off

As shown in the previous experiment, KnowledgeRank can effectively filter out some topics from the knowledge source but cannot perform this task perfectly. Some filtered out topics could have potentially been used to generate the corpus. A natural question to ask is: Should preprocessing be performed at all? Since keeping all topics into the topic model allows the topic model to determine whether the topic is needed based on an accurate Gibbs sampling—this may lead to more accurate topic labeling and better topic interpretability, as it fits with the original model in Section 4.1.3.

The primary factor in deciding to use preprocess filtering is the amount of time it takes a weakly-supervised topic model to run entirely. Figure 4.8(f) signifies this time to be significant for existing weakly-supervised methods with a large corpus, approximation steps, and knowledge source size. This result is validated again in Figure 4.17(c), for a corpus of 2,000 documents averaging 500 words per document and K set to 100 topics and 10 approximation steps, as the knowledge source increases, so does time. At the extreme end, one iteration takes over 350 seconds. It is simply not feasible to run the model on such an input size.

The solution is to reduce the knowledge source size using KnowledgeRank. But by doing this, we sacrifice some F-score. Figure 4.17(c) shows the trade-off expected when we filter out all but K topics from the knowledge source before inference. As expected, as we increase the number of filtered topics, we inevitability decrease the F-score, as the ranking model has more choices to skew the filtering. This relationship is verified with the other datasets in Table 4.13. The anti-correlation (r) is shown in Table 4.13 as the Pearson correlation coefficient.

4.3.3.5 Inference Pruning

Given that the input into the weakly-supervised topic model is a superset of topics, at some point, the topic model must decide which topics to keep and which topics to discard. Additionally, since K unlabeled topics are thrown into the mix in the mixed models, a determination must also be made on these unlabeled topics.

KnowledgeRank can be used in these determinations by helping sort out which topics are best to keep around in a more in-depth manner than the current method of counting. The following experiment verifies this, and compares its selections against clustering-based methods.

4.3.3.5.1 Experimental Setup

A corpus was generated consisting with 2,000 documents having an average of 500 words per document using 100 Wikipedia articles taken from MeSH subject headings. The Source-LDA generative algorithm was used to create the corpus from the 100 selected Wikipedia articles. The parameters for Source-LDA were α, μ, σ set to 0.5, 0.7, and 0.3, respectively. We run Source-LDA with a knowledge source of 1,000 medical subject headings with the generated corpus, inclusive of the 100 selected topics to generate the corpus. This process does not always yield incorrect decisions from Source-LDA using simple counting. Therefore, random permutations of the 1,000-topic superset and 100 selected topic set were used as input into this process. The 100 and 1,000 topic sets were sampled from a full MeSH and UMLS overlapping set of 8,000 topics. Once a corpus and knowledge source were found, we log the decisions, count vectors, and ϕ distributions at each relevant step of the topic model and run the different methods to see if they can improve upon the decisions.

The decisions are made using KnowlegeRank and the established clustering algorithms: k-means clustering and density-based spatial clustering of applications with noise (DBSCAN) [EKS96]. For KnowledgeRank, a graph was constructed using the counts as a weight from a word node and a topic node. If word i was assigned to a topic j, then the number of times that word i was assigned to topic j becomes the weight of the directed edge from node i to node j. These rankings were then used to weigh the counts to decide which topics to keep. K-means and DBSCAN were run against the ϕ distributions. The number of centroids for k-means was set to 100. For DBSCAN ϵ was set to 0.115 with the minimum number of points for a dense region as 1. We take the point closest to the centroid (Centroid), the distance to the centroid (Distance), and the distance weighed using the ranking score (Distance+Rank) to rank and choose topics to keep for k-means. For DBSCAN, we take the topic with minimal distance to all other topics in the cluster. We then perform the same experiment on all non-hierarchical datasets.

4.3.3.5.2 Experimental Results

The algorithms were run against the ϕ and count matrices after 800 iterations of Gibbs sampling. The decision to make is to choose the best 100 out of 176 (this can be different depending on the data source and random seed) candidate topics. As shown in Figure 4.18 and Table 4.14, KnowledgeRank improves upon the existing method of counting, while k-means based decisions and DBSCAN mostly have no effect. From an intuitive perspective, this problem is well served for KnowledgeRank. The reason why the topic model does not assign the words to the correct topic is due to another topic, which is not used in the generation of the corpus, that takes the assignments. By ranking the counts to topics, we can give less importance to words that belong to many different topics. These words can skew the counts and lead to incorrect topic decisions—while weighting them appropriately by the amount they overlap, which ranking methods are quite good at, allows for a better decision.

	Rank-LDA		**Source-LDA**		**EDA**		**CTM**	
	Γ	$\Lambda\%$	Γ	$\Lambda\%$	Γ	$\Lambda\%$	Γ	$\Lambda\%$
MeSH	**935.8**	**62.9**	4432.5	50.1	20390.8	42	2040.7	30.7
PhySH	**47.6**	**75.9**	7262.3	50.2	14447.9	41.5	981.2	39.2
ACM-2012	**3.6**	**49.9**	7637	49.9	1481.9	49.9	200.6	49.9
OAD-Wiki	**202.3**	**74.4**	25407.6	51.8	11363.1	37	1197.9	12.5

Table 4.15: The classification accuracy of token assignments and perplexity values for Rank-LDA, Source-LDA, EDA, and CTM using a corpus mixed evenly between parent and child topics.

	Rank-LDA		**SawETM**		**VRTM**		**VRTM+W2V**	
	Γ	$\Lambda\%$	Γ	$\Lambda\%$	Γ	$\Lambda\%$	Γ	$\Lambda\%$
MeSH	**935.8**	**62.9**	6212.5	1.82	985.1	1.84	2073	1.85
PhySH	**47.6**	**75.9**	2200.6	5.72	910.3	5.72	266.3	5.7
ACM-2012	**3.6**	49.9	326.8	**50.1**	167.4	50	63.6	50
OAD-Wiki	**202.3**	**74.4**	3223.5	3.04	1225.9	3.04	1401.6	3.04

Table 4.16: The classification accuracy of token assignments and perplexity values for Rank-LDA, SawETM, VRTM, and VRTM+W2V using a corpus mixed evenly between parent and child topics.

4.3.3.6 Partial Knowledge

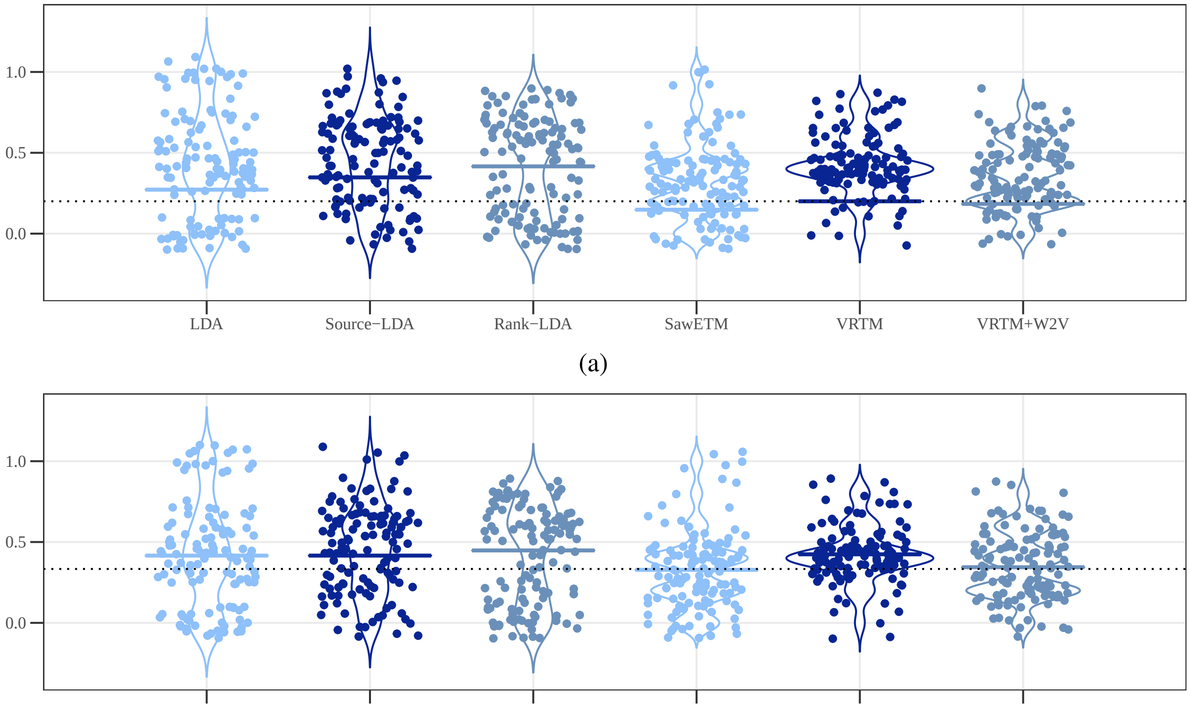
We show the utility of Rank-LDA to aid existing weakly-supervised topic models with partial knowledge. In this experiment, Rank-LDA is used to assign meaningful labels to topics that contain a large number of words that appear in the corpus but do not occur in the predetermined article-based knowledge sources. Each corpus used in evaluation consists of a subset of topics as the knowledge source and their children as the source for tokens not in the knowledge source.

4.3.3.6.1 Experimental Setup

To demonstrate how we constructed the dataset we use the MeSH corpus as an example. All other datasets were similarly constructed. From the entire MeSH hierarchy, we crawl Wikipedia for the resultant topic documents. This set consists of 20,050 found topics. We randomly select 65 topics that contain at least two direct children in the found topics from this set. We take two random direct children from the 65 parent topics for a total of 130 child topics. We then build the knowledge graph based on SNOMED CT filtered by tokens appearing in the corpus. We connect the nodes together using the graph structure of SNOMED CT and the article-based knowledge source. We add an edge between every topic node and a word existing in the corresponding article. Next, we run a modified version of the generative model for the bijective model given in Section 4.1.1 with parameters of K, α β, μ, σ, D, D_{avg}, as 65, $50/K$, $200/V$, 5.0, 0.0, 2,000, and 500 respectively. The modification is for each word we flip an unbiased coin to decide if we are to sample from the parent topic under the Source-LDA parameters or from the raw child distributions. This process results in close to a 50/50 split between a word coming from the parent or the child. For each child word, we mark the topic assignment as that of the parent and keep the parent word as assigned to the parent topic. We then run the Rank-LDA topic model in comparison with Source-LDA, explicit Dirichlet allocation (EDA) [Han13], the concept topic model (CTM) [SSC11], Sawtooth Factorial Topic Embeddings Guided Gamma Belief Network (SawETM) [Dua21], the Variationally-Learned Recurrent Neural Topic Model (VRTM) [Rez20], and a version of VRTM defined to utilize outside information in the form of word embeddings (VRTM+W2V) [MSC13] (evaluated as a separate model) to determine which topic each word belongs to. Each neural topic model was implemented as described in their respective publications [DB21, CBG09]. For all hierarchical datasets, we repeat the experiment as described above with the corresponding dataset topics.

4.3.3.6.2 Experimental Results

After 1,000 iterations, we compare the perplexity (Γ) and classification accuracy (Λ) as a measure of goodness between the models. As is shown in Table 4.15, Rank-LDA outperforms all other weakly-supervised topic models in terms of correctly assigning each word to the correct label ($\Lambda\%$), as well as perplexity. Rank-LDA similarly outperforms the baseline neural topic models as demonstrated in Table 4.16. The weakly-supervised baseline methods, Source-LDA, EDA, and CTM are limited in being restricted by their knowledge source distributions leading to a low probability of a word being assigned to the topic when it is not in the knowledge source topic. Rank-LDA rectifies this deficiency by bringing in additional outside information to connect words that may not show up in the original knowledge source article. Another interesting aspect is that Rank-LDA outperforms the neural topic models in terms of perplexity. It is somewhat expected for Rank-LDA to outperform the baseline models in label assignment accuracy, however perplexity is a major benefit of the



(a)

(b)

Figure 4.19: Results showing mean group answers for the word intrusion task (a) and topic intrusion task (b).

neural topic model over Bayesian models. We submit the reason for better performance has to do with the benefits of our model (over other weakly-supervised models) coupled with the generated data. These results suggest in data that is generated under a generative model, Bayesian models can outperform neural topic models—a finding that suggests the gains in perplexity to Bayesian topic models in reported studies [DB21, CBG09] may be due to the assumed generative model of the Bayesian topic models.

4.3.3.7 Interpretability

To show the how knowledge source rankings affect interpretability, we follow established crowdsourcing techniques [CBG09] to measure interpretability of our proposed model against baseline models. The two interpretability tasks we measure are topic intrusion and word intrusion.

4.3.3.7.1 Experimental Setup

We extract the Wikipedia article for each MedlinePlus article heading from the MedlinePlus corpus. We add into the knowledge source, articles which are a descendant, ancestor, or no relation to a MedlinePlus article heading according to MeSH. The knowledge source consists of 1,000 articles and 1,000 knowledge source topics. We take MedlinePlus as our corpus, which consists of 961 articles. Next, we run LDA with parameters K, α, β as 100, $50/K$, $200/V$ respectively followed by Source-LDA on the corpus with K, α β, μ, σ, as 100, $50/K$, $200/V$, 1.0, 0.3 respectively for 1,000 iterations. Next, we run a version of Rank-LDA where we filter out 800 topics before inference (Preprocessing) and use ranking to prune topics during inference (Inference Pruning). The parameters for Rank-LDA are the same as Source-LDA with ζ_a and ζ_g both as 0.5. The graph used in Rank-LDA is built from the knowledge source and the UMLS described in the methods section. We repeat the above for ten LDA, Source-LDA, Rank-LDA, SawETM, VRTM, and VRTM+W2V runs. The neural topic models were implemented as described in Section 4.3.3.6. To generate the topic intrusion task, we choose a random run, then a random document from a set of MedlinePlus article headings that do not require specialized medical knowledge. After a document is chosen, we take the two most probable topics from θ and a random selection of the least probable topics as the intruder topic. We present the user with the title of the article and the first 100 words—with the option to view the entire article. After reading the title and article, the user must identify the intrusive topic from the set of 3 topic labels. For LDA, we use the eight most common words as the topic label. In the word intrusion task, we first select the output from a random run from a topic model, then randomly choose a topic (distribution over the vocabulary). From the chosen topic, we choose the four most probable words from ϕ. The intrusive word is taken randomly from the five least probable words from ϕ that are also highly probable words in some other topic. The user is then presented with the topic label and asked to choose the intrusive word from the combined set of four probable and one improbable words. We filter out obscure words and topics for both the topic intrusion and word intrusion tasks.

4.3.3.7.2 Experimental Results

The tasks are placed on Amazon Mechanical Turk.[11] For each task, a total of 75 questions are generated, 25 each for LDA, Source-LDA, Rank-LDA, SawETM, VRTM, and VRTM+W2V. Each task is assigned five workers. After the assignments are completed, we compare how well each model did versus the null hypothesis. For the null hypothesis, we assume a random guess. For the topic intrusive task, Rank-LDA and Source-LDA score a p-value of 0.0249 and 0.0742 with mean values of 0.448, 0.416 respectively. These scores imply significance at the 90% confidence level for both models. For the word intrusive task, we obtain p-values for both Rank-LDA and Source-LDA as less than 0.001 with mean values of 0.416 and 0.348 respectively. While there is not much interpretability gain over LDA for the topic intrusion task, there is a significant improvement in the word intrusion task (mean value of 0.272 for LDA against 0.416 for Rank-LDA). The neural topic models perform poorly on both tasks, more so than the Bayesian topic models. These findings are consistent with recent studies on neural topic models and interpretability [DB21]. Each task's results are plotted as a box plot in Figure 4.19. Each dot represents an answer whose value is set to the mean of that group. The groupings are based on the worker and topic for the word intrusion task, and worker and document for the topic intrusion task. The dashed line represents the mean of the null hypothesis.

4.3.4 Discussion

This work aims to remove a barrier to the widespread use of weakly-supervised topics models. Given that we can now use any size knowledge source size as input that can be run using PageRank, speed of execution is no longer an impediment. Existing weakly-supervised topic modelers may find our approach to be beneficial as well to improve upon existing running times. Additionally, we provide an alternative to counting in the inference stage, which yields better topic elimination decisions. When used as an extension in the generative model itself (Rank-LDA), we provide a topic model that improves the state-of-the art method (Source-LDA) for both perplexity and interpretability. Put together, the techniques presented in this work yields improvements in three

[11] https://www.mturk.com/

vital areas of topic modeling: (1) execution time, (2) perplexity, and (3) interpretability. We hope that by demonstrating these improvements, weakly-supervised topic modeling becomes more widely used.

The improvements of our method over existing weakly-supervised topic models is impactful, however it is far from complete. One limitation is the input size of the weakly-supervised input. Given the nature of PageRank, the input size is limited to the order of 10^5. While this greatly increases the number of inputs that can be handled by our model, it is conceivable that an input size may be larger than 10^5, and unable to be used with out model. Future work may consider a non-parametric model to handle a theoretically infinite input size. Also, the ranking method we provide here may be vastly improved as input into an ensemble technique which uses the baseline methods together with other information retrieval techniques. The ensemble method approach is left as an open research area. Another limitation is the inputs used in weakly-supervised topic models. While generally less restrictive than supervised learning models, there is a limited amount of weakly-supervised data available. In our study we utilize Wikipedia due to its completeness, but outside of Wikipedia for the general domain there is not too many alternatives. This hurdle is an interesting area of future research.

Although the addition of graph-based knowledge sources has been established as beneficial to various text mining tasks, it has yet to be demonstrated for inputs used in weakly-supervised topic modeling. Our approach represents a novel technique for representing a knowledge source article as a graph and extracting meaningful information from that graph. We also provide a technique to add additional contextual knowledge into topic modeling. Our work is the first topic model that biases topic construction to both written word knowledge sources and graphical-based knowledge sources. It also represents the state-of-the-art technique for weakly-supervised topic modeling when given both graphical and text-based knowledge sources, in both perplexity and interpretability measurements—and when the knowledge source input size is very large, our method is the only feasible technique currently available.

This work introduces novel methods for representing knowledge sources as graphs and ranking the nodes representing topics. These rankings can be applied to existing weakly-supervised topic models. When used in the preprocessing stage, KnowledgeRank is helpful to eliminate unnecessary topics. Eliminating topics before inference helps speed up the topic modeling and allows the topic model to focus on a more appropriate superset of source topics. This ranking can be used during inference in place of existing elimination techniques based on counting or clustering. When used alongside weakly-supervised models that use an article-based knowledge source, a graph-based knowledge source improves the topic labeling. This also results is better perplexity and improved interpretability.

4.4 The biased coin flip process

To further advance weakly supervised topic models we seek to eliminate some bounds of the process. One such bound is the number of topics, which is unreasonable to assume to be known beforehand. In order to make the topic models more dynamic and potentially more interpretable, we integrate Bayesian non parametric theory into our probabilistic model. However, before we introduce our technique of combining weakly supervised topic models with non parametric models, we seek to improve on existing non parametric topic models.

Bayesian nonparametric learning is a form of Bayesian learning that involves model inference without some traditionally used parameter(s). For example, in topic modeling, it is assumed that the number of topics is given as input beforehand. If we would like to discover topics for a corpus without knowing the number of topics a priori, an applicable model for topic discovery would be a model which utilizes Bayesian nonparametric learning. In nonparametric topic modeling, often some form of the Dirichlet process is used to infer topics. The Dirichlet process is not only useful for topic modeling but also useful for many learning tasks where some set of input parameters are unknown [DMG20, KRV20, SLN20, CJ88].

As discussed above, nonparametric learning is advantageous when some set inputs parameters are unknown. One naive solution to finding the correct set of inputs is to use a brute-force method to try every possible or probable input parameter. Then after the model is learned, some type of scoring metric would need to be used to compare different runs. This however is, if not intractable, then extremely inefficient. The Dirichlet process rectifies this by allowing for an infinite set of possibilities—all while performing inference in a relative efficient time.

The Dirichlet process can discover predictions using an infinite set of inputs by gradually decreasing the probability a new input parameter is expanded. In the context of topic modeling, this takes on the form of the number of topics. During inference, a new topic is created with a decreasing probability. Though it is theoretically possible for an infinite number of topics, each new topic decreases the chance of creating another topic so that the number of topics in practice converge to a finite number.

The Dirichlet process is often expanded to include two Dirichlet processes, with one being an input to the other in what is known as a hierarchal Dirichlet process (which is a different concept altogether from hierarchal topic modeling [Ble03]). The hierarchal Dirichlet process is the distribution used in the theoretical generative model for nonparametric topic modeling. This process complicates the original non hierarchal process in it's generative model and inference calculation. In nonparametric topic modeling inference is done using approximation techniques [Hei11, Wal08] or more esoteric techniques to sampling [TGG07, IJ03]. The approximation techniques are limited to how well the approximation fits the underlying calculation—which can lead to the possibility of less-than-optimal results. Additionally, the more esoteric sampling techniques require a greater cost to understand the material.

Are the existing methods serving to impede adoption of nonparametric Bayesian topic modeling? Indeed, it does appear that in topic modeling, parameter based Dirichlet methods are more popular than Dirichlet process methods[12]. In this paper we seek to improve the inference capability of existing nonparametric topic modeling leading to better predictive performance. By changing a slight yet fundamental detail in how the process generates data, we can frame the inference as a sub routine of the already adopted, well documented, and less complex latent Dirichlet allocation inference [BNJ03]. To help understand the intuition behind the new inference calculation it helps to interpret the Dirichlet process in a new light, in what we introduce as the biased coin flip process.

[12] Based on Google Scholar index of research publications in 2019

4.4.1 Methods

We describe the biased coin flip process in the context of a bank deciding how to partition a set of coins. To begin with, a teller at the bank sets aside an infinite number of bags labeled numerically starting from 1. Each bag is also associated with a coin flip bias h, and a sample from a distribution G_0. Additionally, we assume there is a row of coins on a table and there exists a way to assign a uniform bias to each coin on the table. The process begins at the first bag, B_1. The bank teller takes the bias associated with B_1, h_1 as the bias to make each coin on the table flip heads with probability h_1. Next, the teller takes the first coin on the table and flips it. If the coin lands on heads, the coin is placed in bag 1 (B_1). If the coin lands on tails the coin is placed back on the table, never to be flipped again in this initial step. After the flip of the first coin, the teller moves to the next coin on the table and repeats the process until all coins have been flipped. At this point we say the teller is done with bag 1, then proceeds to take the bias out of bag 2 and sets all the coin's (on the table) biases to h_2. The teller proceeds the same procedure with all coins left on the table. This process is repeated until all coins are off the table. An algorithmic description of this would be as follows:

```
for $i \leftarrow 1$ to $\infty$ do
    Choose $\phi_i \sim G_0$
    Choose $h_i \sim Beta(1, \gamma)$
    $B_i \leftarrow \{\}$
end for
for $i \leftarrow 1$ to $\infty$ do
    for all $c_j \in C$ do
        Choose $f \sim Bernoulli(h_i)$
        if $f = 1$ then
            $C \leftarrow C \setminus \{c_j\}$
            $B_i \leftarrow B_i \cup \{c_j\}$
        end if
    end for
end for
```

At first glance it may not appear as though the binary coin flip process is equivalent to n draws from a Dirichlet process. We prove equivalence below:

$$B = \bigcup_{i=1}^{\infty} \{\phi_i\} \times C_i \tag{4.55}$$

$$C_i^* = C_{i-1}^* \setminus C_i \tag{4.56}$$

$$C_i = \{c_j \in C_{i-1}^* \mid f_{i,j} = 1\} \tag{4.57}$$

$$f_{i,j} \sim Bernoulli(h_i) \tag{4.58}$$

$$h_i \sim Beta(1, \gamma) \tag{4.59}$$

$$\phi_i \sim G_0 \tag{4.60}$$

alternatively, we can write this as:

$$G_n = \sum_{i=1}^{\infty} \phi_i \cdot C_i \tag{4.61}$$

$$C_i = (\overrightarrow{\mathbf{1}} - \sum_{j=1}^{i-1} C_j) \cdot f_i \tag{4.62}$$

$$f_{i,j} \sim Bernoulli(h_i) \tag{4.63}$$

$$h_i \sim Beta(1, \gamma) \tag{4.64}$$

$$\phi_i \sim G_0 \tag{4.65}$$

Where $\overrightarrow{\mathbf{1}}$ is a vector of all 1's. Since $C_{i,j} = 1$ with probability $h_i \cdot \prod_{k=1}^{i-1} 1 - h_k$ we can rewrite the above as:

$$G_n = \sum_{i=1}^{\infty} \phi_i \cdot f_i \tag{4.66}$$

$$f_{i,j} \sim Bernoulli(h_i \cdot \prod_{k=1}^{i-1} 1 - h_k) \tag{4.67}$$

$$h_i \sim Beta(1, \gamma) \tag{4.68}$$

$$\phi_i \sim G_0 \tag{4.69}$$

And thus each $G_{n,j}$ will be ϕ_i with probability $h_i \cdot \prod_{k=1}^{i-1} 1 - h_k$—this becomes a discrete distribution over each ϕ_i, and can be reformulated as:

$$G_{n,j} = \sum_{i=1}^{\infty} h_i \cdot \prod_{k=1}^{i-1} (1 - h_k)\delta_{\phi_i} \tag{4.70}$$

which has previously been established as equivalent to the Dirichlet process [Pai10].

The biased coin flip process is equivalent to the Dirichlet process in the same way that the Chinese restaurant process, the stick-breaking process or the Pólya urn scheme is equivalent to the Dirichlet process. It represents an alternative view. We maintain the benefit of this view is that it guides the thought process of the Dirichlet process away from a single draw to a series of draws (coins on a table). In this way it represents a departure from existing interpretations, such as the stick-breaking process or the Chinese district process [PC09]. Another important difference is this interpretation leads to a novel yet familiar inference calculation. To establish these points more succinctly—the BCP is equivalent to the Dirichlet process, yet represents an alternate view. This alternate view distinguishes itself from existing views and contributes in two important ways: (1) the BCP view frames the process as a series of draws, as opposed to a single draw, and (2) this view allows for inference to be done in a similar way to LDA.

So given an input of $C = c_1c_2c_3 \ldots c_n$ we are tasked to find the matrix z with $z_{ij} \in \{H, T\}$ and ϕ, which will take the form of:

$$p(z,\phi|C, \gamma) = \frac{p(z,\phi,C|\gamma)}{p(C|\gamma)} \tag{4.71}$$

But this looks strikingly like existing Dirichlet process inference equations, and indeed in this form the biased coin flip process is not of much use.

The major advantage of the bias coin flip process occurs when we are asked to find z_t and ϕ_t given $C_t \subseteq C$. From the biased coin flip analogy, this is equivalent to finding the assignments of H and T for the coins left on the table for the tth bag—as well as the tth bag's distribution ϕ_t. If the underlying distribution of G_0 is a Dirichlet distribution parameterized by β, the probability becomes:

$$p(z_i,\phi_i|C_i,\gamma,\beta) = \frac{p(z_i,\phi,C_i|\gamma,\beta)}{p(C_i|\gamma,\beta)} \tag{4.72}$$

Which thought of in a different way is the exact same calculation as finding the topic assignments of a single document with $K = 2$, $\gamma = \alpha = 1$, $w = C_t$ and $V = 2$. In fact a Gibbs sampler (for Equation 4.72) would then be indistinguishable from Equation 4.12.

We now have all the components necessary to build the Gibbs sampler for the entire biased coin flip process. If we assume one time step to be flipping all the coins on the table for a single bag, then for each bag B_t, we calculate the probability of a coin belonging to bag B_t using Equation 4.12 multiplied by the probability the previous $t-1$ flips were all tails. This allows for a recursive multiplication of the previous probabilities that the current coin lands on tails. We can split the head and previous tail probabilities for the ith coin at time t as:

$$P(z_i{=}1|z_{\text{-}i},w_t,p_{t-1}) \propto P(z_i{=}1|z_{\text{-}i},w_t) \cdot p_{t-1} \tag{4.73}$$

and cumulated tail probability as:

$$p_t \propto P(z_i{=}0|z_{\text{-}i},w_t) \cdot p_{t-1} \tag{4.74}$$

To account for an infinite amount of time steps we consider the bag that contains the very last coin from the table (b'). At this point any remaining time step follows a monotone probability calculation of:

$$P(z_i{=}1|z_{\text{-}i},w_t,p_{t-1},t) \propto \frac{1}{V} \cdot \frac{1}{K\alpha} \cdot p_{b'} \cdot p_e^{t-1-b'} \tag{4.75}$$

and p_e as:

$$p_e \propto \frac{1}{V} \cdot \frac{\gamma}{K\alpha} \tag{4.76}$$

To aggregate the mass for all bags $t > b'$ we can take the improper integral which equates to:

$$\int P(z_i{=}1|z_{\text{-}i},w_t,p_{t-1},t)dt \tag{4.77}$$

This can be further simplified by normalizing the posterior conditionals when $t > b'$

$$P(z_i\text{=}1|z_{\text{-}i},w_t) = \frac{P(z_i\text{=}1|z_{\text{-}i},w_t)}{\sum P(z_i|z_{\text{-}i},w_t)} = \frac{1}{1+\gamma} \tag{4.78}$$

and

$$P(z_i\text{=}0|z_{\text{-}i},w_t) = p_e = \frac{\gamma}{1+\gamma} \tag{4.79}$$

with the total mass as:

$$\int P(z_i\text{=}1|z_{\text{-}i},w_t,p_{t-1},t)dt = \frac{\text{-}p_{b'} \cdot p_e}{\gamma \cdot \ln(p_e)} \tag{4.80}$$

4.4.1.1 Hierarchical Biased Coin Flip Process

In the hierarchical biased coin flip process, we choose a biased coin flip process as the base distribution. The "parent" process then will take a Dirichlet distribution as its base distribution. We can extend our bank analogy by adding a central branch. The bank teller at the local branch continues with setting aside an infinite number of bags. But instead of generating the distributions for each bag, the teller must call the central branch to get the distribution. For each call to the central branch, the bankers place a coin on its table. And before any of the local branches were even created, the central branch had already generated an infinite number of bags, each with their associated biases and draws from the base distribution G_0.

In a topic modeling analogy each local branch represents a document, and each local branch coin is a word. To the central branch each bag is a topic (ϕ_i), and each coin is a bag of words corresponding to a particular topic assignment in a particular document (θ_{ij}). To describe it formally:

```
$C^* \leftarrow \{\}$
for $i \leftarrow 1$ to $\infty$ do
    Choose $\phi_i^* \sim G_0$
    Choose $h_i^* \sim Beta(1, \zeta)$
    $B_i^* \leftarrow \{\}$
end for
for $j \leftarrow 1$ to $D$ do
    for $k \leftarrow 1$ to $\infty$ do
```

```
        $C^* \leftarrow C^* \cup \{c_{jk}\}$
        Choose $h_{jk} \sim Beta(1, \gamma)$
        $B_{jk} \leftarrow \{\}$
    end for
end for
for $i \leftarrow 1$ to $\infty$ do
    for all $c_{jk} \in C^*$ do
        Choose $f \sim Bernoulli(h_i^*)$
        if $f = 1$ then
            $C^* \leftarrow C^* \setminus \{c_{jk}\}$
            $B_i^* \leftarrow B_i^* \cup \{c_{jk}\}$
            $\phi_{jk} \leftarrow \phi_i^*$
        end if
    end for
end for
for $j \leftarrow 1$ to $D$ do
    for $k \leftarrow 1$ to $\infty$ do
        for all $c_l \in C_j$ do
            Choose $f \sim Bernoulli(h_{jk})$
            if $f = 1$ then
                $C_j \leftarrow C_j \setminus \{c_l\}$
                $B_{jk} \leftarrow B_{jk} \cup \{c_l\}$
            end if
        end for
    end for
end for
```

In the hierarchical biased coin flip process, the Gibbs sampler equations remain the same except for the conditional posterior at the central branch level. This must consider the word bag instead of a single token. This reduces to:

$$P(z_i{=}j|z_{\text{-}i},w) \propto \prod_{w_i \in B_i^*} \frac{n^{w_i}_{\text{-}B_i^*,j} + \beta}{n^{(\cdot)}_{\text{-}B_i^*,j} + V\beta} \cdot \frac{n^{d_i}_{\text{-}i,j} + \alpha_j}{n^{(d_i)}_{\text{-}i} + K\alpha} \tag{4.81}$$

To calculate the infinite mass sum, the only changes to Equation 4.80 are to p_e

$$p_e = \frac{\zeta}{1 + \zeta^{b'}} \tag{4.82}$$

With ζ being the scaling parameter at the central branch level.

	Description	**D**	**K**
CiteULike-180	Manually tagged scholarly papers	182	1,660
SemEval-2010	Scientific articles with manually assigned key phrases	244	3,107
NLM500	A collection of PubMed documents and MeSH terms	203	1,740
RE3D	A set of labeled relationship and entity extraction documents	98	2,200
Reuters-21578	Manually labeled documents from the 1987 Reuters newswire	21,578	2,700
Wiki-20	20 Computer Science papers annotated from Wikipedia articles	20	564
FAO-30	Manually annotated documents from the Food and Agriculture Organization of the UN.	30	650

Table 4.17: Datasets used in the evaluation of the biased coin flip process and a description of the number of documents in the coporus (D) and topics (K) used in the corpus.

4.4.2 Evaluation

Having already proved the theoretical equivalence to the Dirichlet process, we seek to do the same empirically. Since the biased coin flip process does not rely on approximations, it is entirely possible that it can outperform existing methods. We show this is indeed the case in terms of predictive ability. Next, we show the ability to accurately discover the correct number of topics given a generated corpus. Finally, we show the general applicability of the BCP in a task aimed at discovering the parameters used in a Gaussian mixture model (GMM). The datasets used in evaluation are described by Table 4.17.

4.4.2.1 Perplexity

We seek to compare the ability of the biased coin flip process to predict held out data against the established methods of: the Infinite-LDA (INF) [Hei11], the Nonparametric Topic Model (NTM) [Wal08] and the hierarchical Dirichlet process [TJB06].

4.4.2.1.1 Experimental Setup

For all models we set the hyperparameters for γ as 1, and β as $200/V$, with V being the size of the vocabulary. We then run each topic model for 1000 iterations. Each data set was cleaned to strip out stop words, excess punctuation and frequent and infrequent terms. Additionally, since all baseline models can update their respective hyperparameters during inference [TJB06, EW95], we add these models to our baseline comparison. For the perplexity analysis we take roughly 80% of the corpus for training and test on the remaining 20%.

4.4.2.1.2 Experimental Results

After the 1000 iterations had completed, we compare the ability for each model to predict the held-out data. We calculate perplexity to be:

$$\sqrt[T]{\prod p(w_i|D_t)} \tag{4.83}$$

With T the sum of all tokens in the test corpus, w_i the word at token i in document D_t.

As we show in Table 4.18 and Table 4.19, out of all the models the BCP performs the best by a substantial amount. We hypothesis this is due to a more direct inference calculation that considers two sets of concentration parameters: one for the local and central branch. This surprising result emphasizes the importance of the inference calculation when performing nonparametric topic modeling. Additionally, we find that optimization does have much of an effect. In some datasets predictive power is better, while in others it is worse.

4.4.2.2 K-Finding

We propose a way to test the topic discovery capability of each model is to generate a document using the hierarchical Dirichlet process's generative model and recording the number of topics generated (K). Then we compare the found number of topics for each model run on the generated dataset.

4.4.2.2.1 Experimental Setup

For each dataset we take a histogram of the words as the Dirichlet hyperparameter input for a new topic to be created. We set the corpus size to 1000 documents and take the average document size as a sample from the Poisson distribution having a Poisson centering parameter of 100. With the sampled number of words, we sample from the hierarchal Dirichlet process to get a topic distribution. We then sample a word from the returned topic distribution. This process is continued for all 1000 documents. We repeat this corpus generation process for different values of γ and ζ, each ranging from 0.1 to 4.0. Additionally, we consider a model to "timeout" if the number of discovered topics exceeds 1000. At this point a heat map score of 0 is assigned to the run. We do this because at extreme topic counts, the computation time becomes infeasible for the current implementation of the models.

	BCP	**Inf-LDA**	**NTM**	**HDP**
CiteULike-180	**2262**	36742	71244	13873
SemEval-2010	**6591**	64963	54397	92635
NLM500	**4306**	54333	66652	20846
Re3d	**134**	312	323	1492
Reuters-21578	1591	1168	**860**	3160
Wiki-20	**337**	1645	2152	3406
FAO-30	**314**	4353	4143	5878

Table 4.18: Perplexity of the biased coin flip process (BCP) compared against baseline methods.

	BCP	**Inf-LDA-Opt**	**NTM-Opt**	**HDP-Opt**
CiteULike-180	**2262**	40695	68435	81111
SemEval-2010	**6591**	43192	49309	92635
NLM500	**4306**	46870	69815	20846
Re3d	**134**	312	366	7834
Reuters-21578	**1591**	1890	900	3093
Wiki-20	**337**	2087	1844	20582
FAO-30	**314**	6391	6766	16578

Table 4.19: Perplexity of the biased coin flip process (BCP) compared against baseline methods with optimized parameters.

4.4.2.2.2 Experimental Results

Along with the bias coin flip process and the two baseline models, we also run the two baseline models with parameter updating. Each model is run with the scaling parameters equal to what generated the corpus. We present the results as a heat map, shown by Figure 4.20. To calculate the heat map values (M), we define a metric of similarity that must account for up to an infinite distance from the true value (K). We use the sigmoid function to map the negative K, positive infinity range into the interval $[0, 1]$. However, we want to want to reward answers that are close to the target value more so then answers that are extremely far. The sigmoid function is too sensitive at values close to 1 and quickly jumps to the outer bounds at higher values. For this we take the difference as a percentage of the target value. We formulate this as:

$$E_{\hat{k}} = \frac{|K - \hat{k}|}{K} \tag{4.84}$$

$$M = 2 \cdot \left| \frac{\text{-}1 + exp(\text{-}E_{\hat{k}})}{2 + 2 \cdot exp(\text{-}E_{\hat{k}})} \right| \tag{4.85}$$

This trivial example underscores some of the difficulties in using previous hierarchical Dirichlet processes. We would expect each model to discover the K topics within a reasonable error. However, as we can see from the heat map, only BCP reliably does. Inf-LDA has the tendency to increasingly add topics, making the error from the target larger as the number of iterations increase.

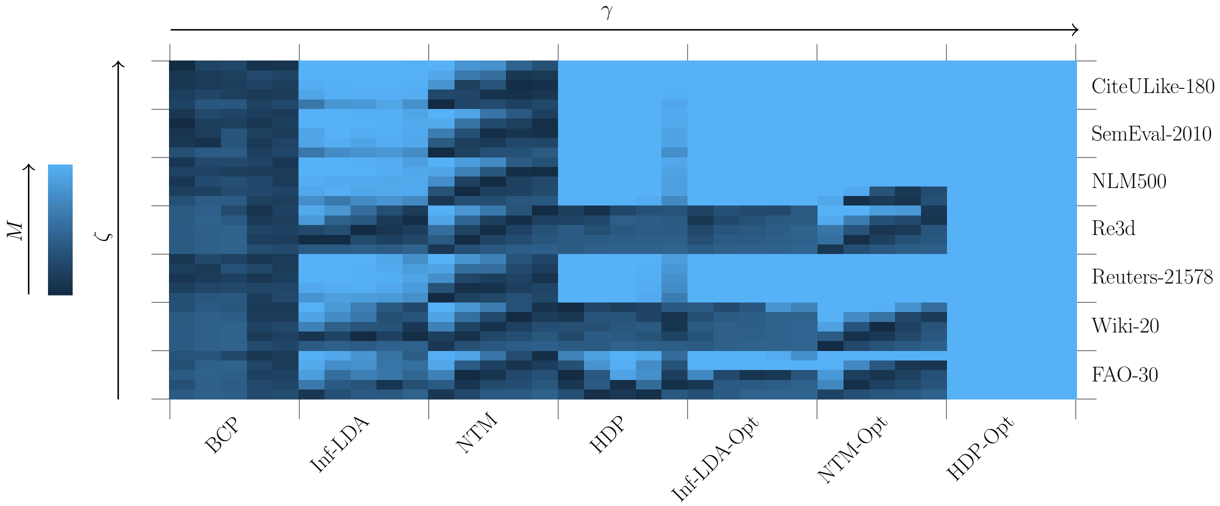


Figure 4.20: Heat map showing the error in finding K topics.

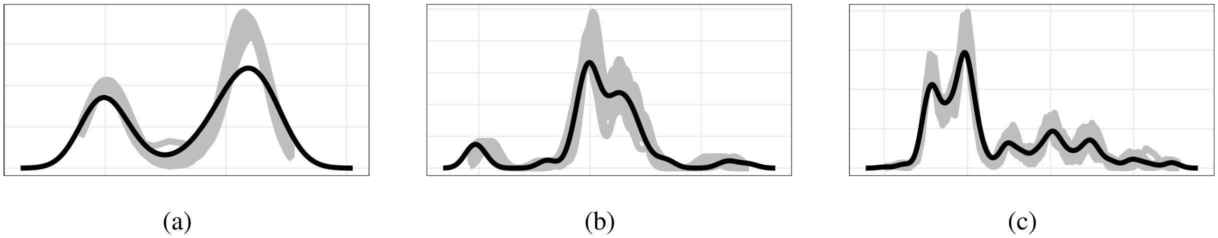
Likewise, this increasing topic effect happens with the Nonparametric Topic Model outside of the diagonal—though not to the same effect as Inf-LDA. Much like the perplexity results, it is the author's intuition that a more direct inference calculation is leading to superior results. It may also be to the act of including both scaling parameters—as the biased coin flip process uses the same amount as the stick-breaking process—which was how the corpora were generated. It does appear that for NTM, when scaling parameters were the same the results improve. However, for Inf-LDA this is not the case. Additionally, the parameter updates should rectify this deficiency but fail to do so.

4.4.2.3 Gaussian Mixture Models

To test the BCP on non topic modeling tasks we seek to find accurate predictive densities of real-world data sets. These datasets are assumed to be generated from a mixture of Gaussian distributions. The three datasets used in analysis are Faithful [AB90], Stamp [IS88], and Galaxies [PHG86]. For each dataset we follow established techniques [IJ02, McA06] of using our nonparametric model to estimate a Gaussian mixture density. We assume a fixed variance and take 30 samples of the mixture density at various iterations after an initial 1,000 iterations. The densities of the samples from the BCP are plotted against their respective kernel density estimates in Figure 4.21. As we can see from Figure 4.21, the densities from our model closely resembles that of the density from kernel density estimation. This similarity suggests our model to be useful in tasks outside of topic modeling such as estimating Gaussian mixture models.

4.4.3 Discussion

This work introduces a novel way to think about Dirichlet and hierarchical Dirichlet processes. Thinking about the process as a series of coin flips, where each round we partition the coins into bags and what's left on a table, we can see the similarity to an established method for inference—latent Dirichlet allocation. Because this method is based on topic modeling inference, it may lead to better results in the context of discovering topics.



(a) (b) (c)

Figure 4.21: 30 density estimates taken from the BCP shown in gray plotted against a kernel density estimation for the Faithful (a), Galaxies (b) and Stamp (c) datasets.

The downside of the technique presented in this paper is the increase in execution time. Since we are performing two sets of Gibbs sampling, one at the local branch level and one at the central branch level, we ultimately need more computations than baseline methods. It is left as an open research area to find improvements. Although not implemented for the biased coin flip process, it may be possible to apply a concurrent processing approach [WTW17, WBS09, NAS07, PNI08], such as that given in Section 4.1.3.4, on the different branch levels. Additionally, an interesting area to investigate would be optimization of the two scaling parameters. This would then allow for the model to be completely parameter free.

We maintain that the downsides of our approach are outweighed by the upsides. The reliance on previous established parametric topic modeling inference calculations leads to a theoretical advantage to existing techniques. And as we show empirically, the biased coin flip process does seem to yield better results. In both prediction of held out data and finding the appropriate number of topics, our model improves upon existing methods.

4.5 The intepretable topic model

The previous sections have all established a context that leads us to the development of a self-contained highly interpretetable topic model with topic labels. This newly developed model resolves all of the shortcomings in previously discussed weakly supervised topic models. Additionally, we eliminate the need to supply a knowledge source as input and specify the number of topics. From an input of just a set of hyperparameters and a corpus, we present a technique to discover highly interpretable topics and topic labels in an efficient manner.

As we have previously shown, topic modeling is an effective way to analyze unstructured textual data. Even with the emergence of the neural topic model, the most prominent technique (based off citation count) for topic discovery is based off a Bayesian graphical model that utilizes the Dirichlet distribution for the inference of topics [BNJ03]. The basic assumption of these models consist of a generative model for the input text. Words are generated by first sampling a topic assignment from a document-level topic distribution. Then for the topic assignment a word is generated from the corresponding topic-level word distribution. This process is completed over the entire length of the corpus. Inference then is done using some Bayesian inference techniques such as Gibbs sampling [GS04, Gri02].

The topics themselves consist of word assignments from the corpus to the topic. The word assignments are then clustered together to form a topic. The topics become just a list of word assignments, i.e., there is no single n-gram that describes the topic. A word list represents a divergence from how a layman might think of what a *topic* is— which could be: *the subject of a discourse or of a section of a discourse* [Top21]. This divergence is at the center of interpretability. Interpretable topics bridge this gap by providing the cluster of words with a single label that serves to best explain the topic (i.e., all words are semantically connected to a notational label). For example, if the top 3 words for a topic are: *pitcher, batter, and outfielder* an interpretable topic for this topic could be *baseball*—which could easily match what a layman would say the topic is given the top words. It is important to note that interpretability is more focused on forming a cluster of semantically connected words, than about topic labeling (however this is a tangential concept).

However, it has long been established that existing topic models fail in trivial aspects of interpretability [CBG09]. Even though the traditional topic modeling methods do not provide a label comprising their most popular word assignments, one would assume there to be be a semantic coherence. But this is not always the case [CBG09]. As we demonstrate in Case Study 4.0.0.1, a major reason for this is that the models tend to assign words to the same cluster (topic) that occur together rather than being semantically connected. For our baseball example, this may lead to a topic discovered that contains the words: *carrots, batter, and galaxy*. For this example, it is hard to place a single n-gram over the topic. From an intuitive perspective this is not unexpected given the nature of the generative model. No condition is placed upon the words to assure semantic relatedness.

Non-parametric topic models do not serve to resolve the deficiencies of interpretability. They do however allow for topic models to be defined over an infinite parameter size. Additionally, they do not require certain parameters to be known a priori. In non-parametric topic modeling the parameter that is left out is often the number of topics. This is advantageous since it somewhat unreasonable to assume the known number of topics a generative model used to create a corpus. Often, traditionally used numbers are used (100 [BNJ03, RGS04]) by default without much analysis of different topic numbers. And to evaluate models learned with differing number of topics, with a log-likelihood comparison for example, is too time consuming and thus different topic number consideration is often discarded.

The technique to non-parametric topic analysis is usually based off the Dirichlet process which bears a resemblance to the Dirichlet distribution. The Dirichlet process specifies a technique to generate a distribution that relies on an infinite number of steps. The main components consist of an underlying distribution and a partition of the infinite probability space to returning a sample from the underlying distribution. Each partition is assigned a sample from the underlying distribution and subsequent samples to the Dirichlet process contain a technique to search existing partitions to return their assigned sample, or create a new partition with a new sample from the underlying distribution. Most often the underlying distribution is a Dirichlet distribution.

Non-parametric topic modeling consists of a "child" Dirichlet process where the underlying distribution is a "parent" Dirichlet process. The "parent" Dirichlet process then uses a Dirichlet distribution as its underlying distribution. This arrangement of "child" and "parent" Dirichlet processes is sometimes called a hierarchical Dirichlet process (not to be confused with hierarchical topic modeling [Ble03]). Existing methods have been shown to be effective in discovering topics sans the number of topics as input. However, non-parametric topic models are not as widely adopted as parametric-based topic models[13].

The connection between non-parametric topic modeling and interpretability lies with weakly-supervised topic modeling. Weakly-supervised topic models concern themselves with assigning labels to topics. By consequence of their method, they also shape the discovered topics to their weakly-supervised topic. Weakly-supervised topic models differ from previous approaches that seek to assign a topic label after inference [LGN11, MSZ07, MMZ12, SXW15, MCN13, HHK13, Pec10]. After inference assignment can lead to somewhat uninterpretable topics as the word assignment cluster representing the topic tend to combine semantically different words. Another approach is to utilize supervised topic labels as input. However, the requirement of an accurate labeled input can be expensive or time consuming to obtain. A fusion of these two approaches are advanced by weakly-supervised input—which allows for an easier to obtain labeled input set and can help form the topics to the labeled input set. A common weakly-supervised input set involves a *knowledge source* which is a collection of articles that are previously labeled. These articles are then turned into distributions. The distributions are referred to as *knowledge source topics*. The components of weakly-supervised topic models are explained by Case Study 4.3.0.1, and supplemented by the following case study:

Wikipedia knowledge source

If we crawl the textual content of the Wikipedia page for "Cancer" [Wik21], then the beginning of the page content will start with:

Cancer is a group of diseases...

Taking the above text fragment to be the full article for "Cancer", we can create a knowledge source topic by taking a histogram of the document and dividing by the total. For our example the knowledge source topic would be the vector: $[\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}]$ with a word vector mapping of:

[13] Based on Google Scholar index of research publications in 2019

[Cancer,is,a,group,of,diseases]. If we continue this same procedure for a set of known illnesses (perhaps from a list of medical subject headings [MeSH] terms), then the collection of knowledge source topics comprises the knowledge source. The knowledge source is then used as the weakly-supervised input set of the topic model. Some of the knowledge source topics will be used and some will ultimately be discarded. By starting with an easy to obtain set of all possible topics, such as MeSH terms, we can label topics for many different medical corpora that only need to be about a subset of the entire MeSH terms (topics) used in the knowledge source.

In Section 4.3.3.6, we demonstrate that in some cases weakly-supervised topic models lead to better predictive power. This result implies that we may not sacrifice much predictive power when adopting a weakly-supervised approach. Additionally, there is a foundation for interpretability. If we can establish that topics, drawn directly from a confirmed knowledge source are highly interpretable, then it follows that topics discovered by a topic model that are biased by the interpretable knowledge source topics would be interpretable as well.

One drawback of weakly-supervised topic models is knowing how many knowledge source-topics to discover. From Section 4.1.3.3, we can see that the models are not well defined in this matter, resorting to some heuristic for topic elimination during inference. For example, using Gibbs sampling, at the start of inference every possible knowledge source topic is considered. Then as iterations increase, knowledge source topics are eliminated based on a rank of word assignments. This is repeated until the total number of topics reach the specified input parameter K. Ideally, this would have a stronger theoretical foundation and not rely on such heuristics as topic assignment counts. Additionally, the model is such that as the number of knowledge source topics increase, so does the computation time. As is shown in Section 4.1.4.5, at a knowledge source input size of just 1,000, the running time is infeasible.

Hence the context for combining weakly-supervised topic models with non-parametric topic models. If non-parametric models can be defined to reasonably execute for an infinite number of topics, 1,000 topics should be easily attainable. Additionally, we can remove the need to specify the number of known topics beforehand, resulting in a more dynamic topic model. We further extend this combination by removing the requirement to specify the knowledge source a priori.

Upon fusing these two domains together we notice another discovery, that we create a topic model for interpretability. In our method of combining the two models we introduce a parameter that specifies the likelihood a knowledge source biased topic is chosen over a regular topic model biased topic. This parameter acts as a way to increase or decrease the knowledge source topics influence as a percentage. Since knowledge source topics tend to be very interpretable, the parameter then becomes a way to increase or decrease interpretability, by a pre-specified amount. By combining weakly-supervised topic models and non-parametric topic models we now have a way to specify the desired level of interpretability.

4.5.1 Methods

To introduce our technique of combining non-parametric and weakly-supervised topic modeling, we begin with the generative model for the hierarchical Dirichlet process, then append weakly-supervised topic modeling constraints. We then explore various improvements to the challenges our approach entails.

4.5.1.1 Non-parametric weakly-supervised model

Starting with the generative model of a hierarchical Dirichlet process-based topic model of:

$$\theta_d = \sum_{i=1}^{\infty} q_{d,i} \cdot \prod_{\ell=1}^{i-1} (1 - q_{d,\ell}) \delta_{\phi_{d,i}} \tag{4.86}$$

$$q_{d,i} \sim Beta(1, \gamma) \tag{4.87}$$

$$\phi_{d,i} \sim P \tag{4.88}$$

$$P = \sum_{i=1}^{\infty} r_i \cdot \prod_{\ell=1}^{i-1} (1 - r_\ell) \delta_{\phi_i} \tag{4.89}$$

$$r_i \sim Beta(1, \zeta) \tag{4.90}$$

$$\phi_i \sim Dir(\alpha) \tag{4.91}$$

We see that we can easily inject weakly-supervised topic model information into the base distribution ϕ_i. We can simply place a mixture over the alpha-Dirichlet distribution and each labeled topic distribution. If we define B to be the number of all labeled topics, this transforms ϕ_i to:

$$\phi_i \sim M \tag{4.92}$$

$$M = (1 - \xi) \cdot \delta_{\mathrm{A}} + \frac{\xi}{B} \cdot \sum_{i=1}^{B} \delta_{\Omega_i} \tag{4.93}$$

$$\mathrm{A} \sim \mathrm{Dir}(\alpha) \tag{4.94}$$

$$\Omega_i \sim \mathrm{Dir}(\omega_i) \tag{4.95}$$

This newly formulated base distribution allows us to construct the entire generative model. In algorithmic form this would be:

```
**for** $b \leftarrow 1$ to $B$ **do**
    Choose $\lambda_b \sim \mathcal{N}(\mu, \sigma)$
    $\omega_b \leftarrow [(X_{b,1})^{g(\lambda_b)}, (X_{b,2})^{g(\lambda_b)}, \ldots, (X_{b,V})^{g(\lambda_b)}]$
**end for**
**for** $i \leftarrow 1$ to $\infty$ **do**
    Choose $f \sim$ Bernoulli$(\xi)$
    **if** $f = 1$ **then**
        Choose $u \sim$ Uniform$(B)$
        Choose $\phi_i \sim$ Dir$(\omega_u)$
    **else**
        Choose $\phi_i \sim$ Dir$(\beta)$
    **end if**
**end for**
**for** $d \leftarrow 1$ to $D$ **do**
    **for** $j \leftarrow 1$ to $\infty$ **do**
        Choose $r \sim$ Beta$(\zeta)$
        Choose $i \leftarrow 0$
        Choose $f \sim$ Bernoulli$(r)$
        **while** f = 0 **do**
            Choose $r \sim$ Beta$(\zeta)$
            Choose $i \leftarrow i + 1$
            Choose $f \sim$ Bernoulli$(r)$
        **end while**
        $q_j \leftarrow i$
    **end for**
    Choose $N_d \sim$ Poisson$(D^*)$
    **for** $t \leftarrow 1$ to $N_d$ **do**
        Choose $q \sim$ Beta$(\gamma)$
        Choose $i \leftarrow 0$
        Choose $f \sim$ Bernoulli$(q)$
        **while** f = 0 **do**
            Choose $q \sim$ Beta$(\gamma)$
            Choose $i \leftarrow i + 1$
            Choose $f \sim$ Bernoulli$(q)$
        **end while**
        Choose $w_{d,t} \sim$ Multinomial$(\phi_{q_i})$
    **end for**
**end for**
```

With the generative model established we are now able to build a Gibbs sampler for inference. Following established methods, we seek to find the appropriate topic assignment for each token. In our model this takes the form:

$$P(z = i|\beta,\omega,\overrightarrow{w},\xi) \tag{4.96}$$

Each topic assignment is dependent on the assignment of a local stick break ($\hat{q}$), and a mapping of that stick break to the parent stick break ($\hat{r}$). We formalize this as:

$$P(z = i|\beta,\omega,\overrightarrow{w},\xi) = P(M_{\hat{r}}|\beta,\omega,\overrightarrow{w},\xi,\hat{r}) \cdot \sum P(\hat{q} = \hat{r}|\beta,\omega,\overrightarrow{w},\xi,M_{\hat{r}}) \tag{4.97}$$

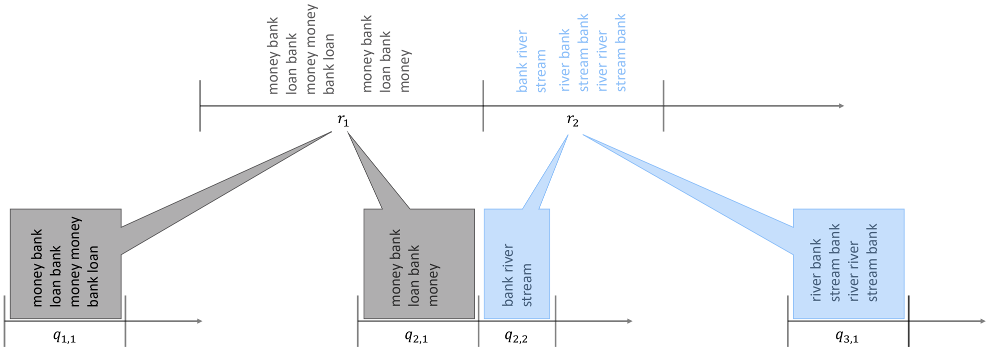


Figure 4.22: A visual representation of a hierarchical Dirichlet process topic model partitioning words into stick breaks.

However, with the change in the underlying distribution this will need to be factored into the posterior distribution and then marginalized out. Letting $\tilde{o}$ be a shorthand for the observables: β,ω,$\overrightarrow{w}$,ξ,$\hat{r}$, our posterior calculation becomes:

$$P(M_{\hat{r}}|\tilde{o}) = (1\text{-}\xi) \cdot P(M_{\hat{r}} = \text{Dir}(\alpha)|\tilde{o}) + \frac{\xi}{B} \cdot \sum_{j=1}^{B} P(M_{\hat{r}} = \text{Dir}(\omega_j)|\tilde{o}) \tag{4.98}$$

The addition of the new underlying distribution does complicate things, but we can reuse existing inference calculations for $\sum P(\hat{q} = \hat{r}|\beta,\omega,\overrightarrow{w},\xi, M_{\hat{r}})$ since this is the basis that every hierarchical Dirichlet process based topic model must calculate. Here we will borrow the calculation from "Infinite LDA" [Hei11] which reduces our calculation to:

$$\sum P(\hat{q} = \hat{r}|\beta,\omega,\overrightarrow{w},\xi,M_{\hat{r}}) \propto p(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i}) \cdot p(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i},\overrightarrow{w},x,y,M_{\hat{r}}) \tag{4.99}$$

$$p(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i},\overrightarrow{w},x,y,M_{\hat{r}}) \propto \begin{cases} \text{Equation } 2.21 & \text{if } M_{\hat{r}} = \text{Dir}(\alpha) \\ \text{Equation } 2.24 & \text{otherwise} \end{cases} \tag{4.100}$$

$$p(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i}) \propto n^{d_i}_{\text{-}i,j} + \gamma \cdot \tau_z \tag{4.101}$$

τ represents a sample from the Antoniak distribution, for further details we refer to the "Infinite LDA" publication [Hei11]. We now only need to marginalize out all the possibilities for $P(M_{\hat{r}}|\tilde{o})$.

With this probability being:

$$P(M_{\hat{r}} = m|\tilde{o}) \propto \prod p(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i},\overrightarrow{w},x,y,m) \tag{4.102}$$

This last probability approximation may seem like a bit of a stretch to the uninitiated. To help the reader understand, we give a simple example and reasoning of the stated probability calculation.

4.5.1.1.1 Bank corpus

We follow a simple and established example involving a corpus containing bank-financial and bank-river terms [SG07]. The corpus involves three documents:

1. money^1 bank^1 loan^1 bank^1 money^1 money^1 bank^1 loan^1.

2. money^1 bank^1 bank^2 river^2 loan^1 stream^2 bank^1 money^1

3. river^2 bank^2 stream^2 bank^2 river^2 river^2 stream^2 bank^2

In our model, perhaps the generative model would partition the words according to Figure 4.22. How would we determine the appropriate $P(M_{\hat{r}}|\tilde{o})$? One can see that this reduces to finding the appropriate topic to a newly introduced problem: the single topic-document model. The single topic-document model is the exact same as a vanilla latent Dirichlet allocation model with the restriction that each document is assigned only a single topic. The single topic is drawn with a likelihood given by the ξ-discrete distribution in $P(M_{\hat{r}}|\tilde{o})$. In more detail:

```
Choose $\phi_1 \sim \text{Dir}(\beta)$
for $t \leftarrow 1$ to $B$ do
    Choose $\lambda_t \sim \mathcal{N}(\mu, \sigma)$
    $\omega_t \leftarrow [(X_{t,1})^{g(\lambda_t)}, (X_{t,2})^{g(\lambda_t)}, \ldots, (X_{t,V})^{g(\lambda_t)}]$
    Choose $\phi_{t+1} \sim \text{Dir}(\omega_t)$
end for
for $d \leftarrow 1$ to $D$ do
    Choose $z_d \sim \text{Multinomial}([1-\xi] \cdot \delta_1 + \frac{\xi}{B} \cdot \sum_{i=1}^{B} \delta_{i+1})$
    Choose $N_d \sim \text{Poisson}(D^*)$
```

```
        **for** $n \leftarrow 1$ to $N_d$ **do**
            Choose $w_{n,d} \sim \text{Multinomial}(\phi_{z_d})$
        **end for**
    **end for**
```

The single topic-document model is uninteresting in itself but one can see the parallel between discovering the probability that a document is assigned topic z_d and determining $P(M_{\hat{r}}|\tilde{o})$. They are reformulations of the same problem. Setting $p(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i})$ from the single topic-document model to the ξ-discrete distribution, we arrive at Equation 4.102.

We now have the basis for non-parametric weakly-supervised topic modeling. We see that we can take an existing non-parametric model and marginalize the underlying distribution representing the weakly-supervised topics. The interesting observation to note is the parameter ξ becomes the likelihood that a weakly-supervised topic is chosen versus a “regular” topic is chosen. If we take a weakly-supervised model to be an interpretable topic (which we will show in the evaluation section) then ξ becomes a parameter specifying the level of interpretability.

4.5.1.2 Knowledge source topic approximation

Discovering topics using a large knowledge source can lead to a severe degradation of execution time. The addition of the weakly-supervised topic model constraints onto the non-parametric Bayesian model imposes a $\mathcal{O}(B \times N_d \times D)$ increase in execution time. One technique to minimize the impact of this time increase is to sample $P(M_{\hat{r}}|\tilde{o})$ at different timesteps than $P(\overrightarrow{z}_i{=}j|\overrightarrow{z}_{\text{-}i},\overrightarrow{w},x,y,M_{\hat{r}})$— such as assigning the appropriate $P(M_{\hat{r}}|\tilde{o})$ at the document timestep as opposed to the token timestep. Another approach we take is to order the most likely knowledge source topics and take only the top s ordered topics. We can then approximate the sum of the remaining $B - s$ topics using an approximation function. If we assume a good ordering, and that each lower ordered function decreases the probability value by a constant, ρ, in the range $(0{,}1)$, then we can calculate the remaining probability as:

$$\mathcal{P}_j^* = P(M_{\hat{r}} = \text{Dir}(\omega_j)|\tilde{o}) \tag{4.103}$$

$$\sum_{i=s}^{B} \mathcal{P}_i^* = \mathcal{P}_{s-1}^* \cdot \int \rho^b db \approx \text{-}\frac{\mathcal{P}_{s-1}^*}{\ln \rho} \tag{4.104}$$

By sampling from this remaining probability chunk we can find the appropriate ordered item.

To order the topics, we examine what makes a good match for a topic. Since we are comparing the bag of words at the parent stick break level, we can use Equation 4.102 to see that words which are assigned to a stick interval a high number of times will tend to match up with a topic that has the same words assigned a high number of times. We can partition each knowledge source topic by each of its top words. Then when we need to get an ordering, we can sort the stick bucket by the top words and search for knowledge source topics that match the bucket's top words. After we acquire a sufficiently sized superset (~ 10 times s) we can order the knowledge source topics using Equation 4.102.

4.5.1.3 Knowledge source discovery

In corpora where a knowledge source is easy to obtain, then we can just use a constructed knowledge source. For example with the Reuters-21578 corpus [reu], we can take the topic label and construct the knowledge source by querying Wikipedia and following the procedure given in Section 4.5. However, many corpora may not have such a simple method to construct the knowledge source.

Our solution involves obtaining the entirety of Wikipedia as a superset of knowledge source topics. We filter out unpopular Wikipedia articles (measured by page views). Because a good match for a knowledge source topic is dominated by token assignments to that topic, it would make sense that words in the corpus that show up in a knowledge source topic many times would be a good fit. We can take simple heuristics for search, such as word count (vote). Word count however can be thrown off by knowledge source topics that contain a lot of words. To account for this we can take word count divided by the total amount of words, or use the established method of term frequency-inverse document frequency (tf-idf) and cosine similarity.

We propose another and novel solution based off existing ranking algorithms [PBM99]. For our knowledge source articles we can model the articles into a graph and then run a ranking algorithm. We take the approach of creating a topic node which is connected to each word in its corresponding source article. This intuition leads us directly to KnowledgeRank, discussed in Section 4.3.

Using the same approach as what was introduced in KnowledgeRank, we use the graph representation and ranking to obtain a subset of knowledge source topics we think would be useful to our model. Additionally, we can use this method in conjunction with other baselines, such as voting.

4.5.1.4 Parameter updating

Due to the Bayesian nature of our model, it may be the case that the ξ guarantee is not met. Ultimately, it will be the data that decides the number of interpretable topics to choose, and ξ will act more as a guide. To enforce a ξ ratio of interpretable topics, we provide techniques for parameter updating.

A simple approach is to use the previous observations of the knowledge source/unlabeled topic ratio to update ξ. If we suppose a linear relationship to the number of topics and ξ, then we can model the expected number of topics given ξ as:

$$E = \mathcal{B}_1 \cdot \hat{\xi} + \mathcal{B}_0 \tag{4.105}$$

The parameters $\mathcal{B}_1$ and $\mathcal{B}_0$ can be updated using linear regression, and $\hat{\xi}$ can be determined by setting E to the total number of topics multiplied by the original value of ξ.

4.5.2 Evaluation

To evaluate the effectiveness of our methodology we set up various experiments. In the first experiment, we test the interpretability of our proposed topic model combination. Next, we show the

	Description	**Documents**	**Topics**
CiteULike-180	Manually tagged scholarly papers	182	1,660
SemEval-2010	Scientific articles with manually assigned key phrases	244	3,107
NLM500	A collection of PubMed documents and MeSH terms	203	1,740
Reuters-21578	Manually labeled documents from the 1987 Reuters newswire	21,578	2,700
Wiki-20	20 Computer Science papers annotated from Wikipedia articles	20	564
FAO-30	Manually annotated documents from the FAO of the UN.	30	650

Table 4.20: Datasets used for evaluation.

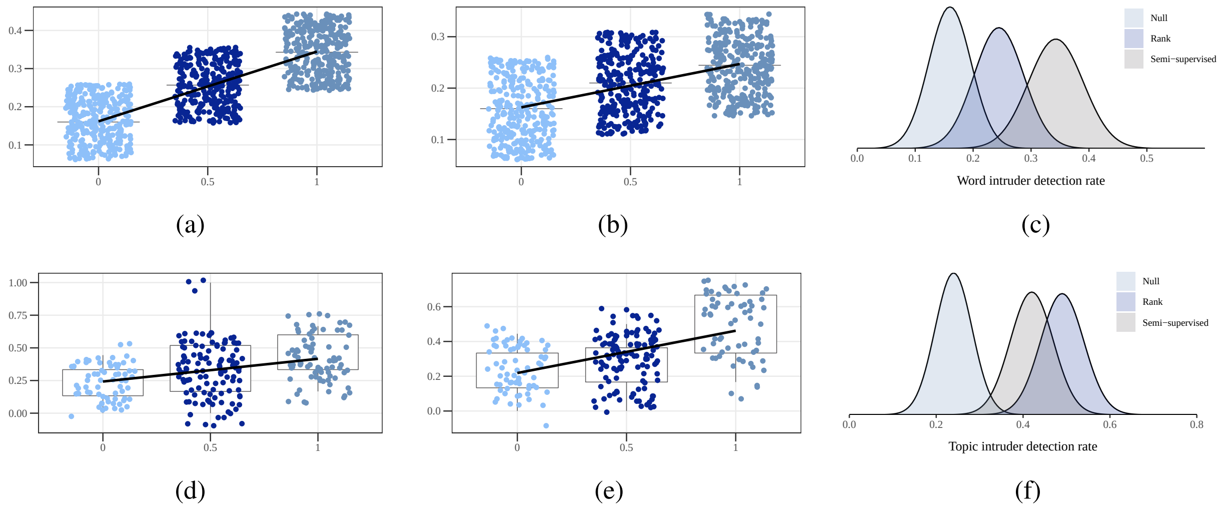


Figure 4.23: Word detection results for all models and datasets with the provided constructed knowledge source (a) and the discovered knowledge source (b). The topic detection task is shown in (d) and (e) for the provided and discovered knowledge sources respectively. The expected distributions for our method at $\xi = 1$ show significance against the null hypothesis distribution for both word detection (c) and topic detection tasks (f).

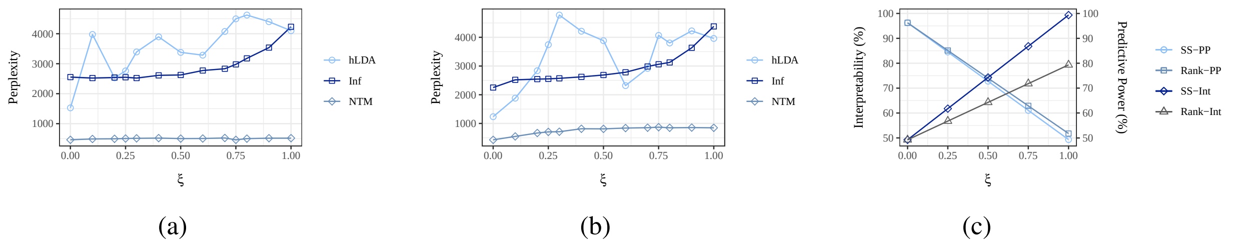
relationship to perplexity, and lastly we evaluate a technique to discover knowledge source topics for a corpus. For all experiments we use the datasets given in Table 4.20. The baseline methods used for our combination method are: Infinite-LDA (Inf-LDA) [Hei11], Hierarchical LDA (hLDA) [Ble03], and the Non-parametric Topic Model (NTM) [Wal08]. For the weakly-supervised portion we adopt the Source-LDA model. For other experiments, we add an additional set of baseline models for comparison with more details given in their respective experiment setups. The weakly-supervised input was taken from the pre-established knowledge source obtained as described by Section 4.5 (Semi-supervised/SS), and a discovered knowledge source described in Section 4.5.1.3 (Rank).

4.5.2.1 Interpretability effect of ξ

We seek to determine how the ξ parameter used in the combination of non-parametric Bayesian and weakly-supervised topic models affects interpretability using human-aided evaluation.

4.5.2.1.1 Experimental Setup

For each baseline model against each dataset, we run the model with the default scaling parameter α as 1 and β as $200/V$ (V being the size of the vocabulary) for 1,000 iterations. After inference was



(a) (b) (c)

Figure 4.24: Perplexity as a function of ξ for the our interpretability method constructed with various baseline models together with the provided knowledge source (a) and the discovered knowledge source (b). The perplexity interpretability relationship is shown as an inverse association (c).

complete, we are able to calculate the document to topic mixture (θ) and topic to word mixture (ϕ) using the end result of the topic assignments. With the θ and ϕ mixtures, we can easily determine the most and least popular word for a given topic, and the most and least popular topic for a given document. We then repeat the same process for all models with weakly-supervised topic modeling appended as described by Section 4.5.1.1. We run the baseline models outside of our interpretability model ($\xi = 0$) against the ξ values of 0.5 and 1. After all runs were completed, we can construct a word intrusion and topic intrusion task to be given for evaluation [CBG09]. Word intrusion involves giving a person 6 words, 5 being from the most popular words in a topic and 1 being among the least popular—the least popular word is referred to as the intrusive word—and asking them to identify the intrusive word. For our evaluation we filter out topics that have 3 or more words that are not in common usage (as determined by showing up in a dictionary word list) or are numeric. Additionally, we restrict intrusive words to the same criteria. We do this because topics which contain all numbers or obscure words may be hard for non-domain experts to understand and so an evaluation with a high percentage of these words might not be meaningful. Topic intrusion is similar to word intrusion only applied to topics. Each user is given a block of text (100 words) that begin a document and are then given 4 topics—3 topics being the most popular in the document and 1 being among the least popular—and asking them to identify the intrusive topic. We utilize Amazon Mechanical Turk as the platform to obtain human evaluation. Each question was given to 3 different Amazon Mechanical Turk users. For subsequent questions, the users were redrawn from the Amazon Mechanical Turk pool of users which reduces the probability that any one single user answered multiple questions.

4.5.2.1.2 Experimental Results

After the users submitted their answers to all questions for the word and topic intrusion task, we evaluated their effectiveness. Each submitted answer was assigned the value of accuracy for its group and plotted in Figure 4.23. The groupings were based on ξ, dataset, and model. We can clearly see the trend between ξ and interpretability for both the topic and word intrusion tasks. In both, ξ is positively associated with interpretability. We show the regression line in each task box plot. Each regression line shows a significance above 0.1. As expected, we see an increase in detection of intrusive words when using the predefined knowledge source versus the discovered knowledge source. However, this is not the case for the intrusive topic. We suppose the topic discrepancies may be due to randomness and does not represent a significant difference. Still, this may represent an interesting point to examine. While the pre-defined knowledge sources are human curated topic labels suggested by reading each document, the discovered ones are more numerical. Numerical in the sense that the only criteria for selecting them are using established methods for information retrieval. It then makes sense that for certain tasks the discovered knowledge source performs better.

Additionally, we calculate whether the models with $\xi = 1$ represent a significant increase in interpretability. The expected distributions, plotted in Figure 4.23(c) and Figure 4.23(f), show significance above 0.1.

4.5.2.2 Perplexity

After establishing the association between ξ and interpretability, we seek to do the same for perplexity. We then aim to link ξ, interpretability and perplexity.

4.5.2.2.1 Experimental Setup

With a total of 3 models, 6 datasets, and 2 sets of knowledge sources (described in Table 4.20 and Section 4.5.2) we vary ξ from 0 (no weakly-supervised input added) to 1. All models had the same scaling parameter of 1, β as $200/V$ and were run for 1,000 iterations. The discovered knowledge source contained 10,000 knowledge source topics and was obtained as described in Section 4.5.1.3.

We also describe knowledge source discovery in more detail in Experiment 4.5.2.3. Before running the topic models on the data, we cleaned the corpora by removing the top 5% of the most and least popular words. For perplexity we split the documents at an approximate split of 80/20.

4.5.2.2.2 Experimental Results

After inference was completed, we calculate the perplexity of the held out data using Equation 4.83. The perplexity is shown as function of ξ in Figure 4.24(a) and Figure 4.24(b). For most models there is a negative association between ξ and perplexity. In other words, we lose predictive power as ξ increases. This is also in line with previous studies [CBG09]. Some models seem to be less affected than others. For example, using NTM with the pre-established knowledge source does not seem to increase the perplexity much. This leads to the interesting possibility of a model that increases interpretability without affecting perplexity. Furthermore, NTM outputs the best perplexity among the models. However, in the aggregate this is not the case. If we take the regression line against all models, we see a negative association. We plot this negative association alongside the interpretability scores reached in Experiment 4.5.2.1 in Figure 4.24(c). We use the metric PP as a function that is just the regression line of the perplexity values with an inverse slope. We can see a general trade-off between interpretability (Int) and perplexity (PP). To gain more interpretability we have to sacrifice perplexity and vice-versa.

4.5.2.3 Knowledge source discovery

In this experiment we test various strategies of discovering a knowledge source for a corpus. We also detail the technique used to construct the knowledge sources utilized in Experiment 4.5.2.1 and Experiment 4.5.2.2.

4.5.2.3.1 Experimental Setup

Starting with the entire Wikipedia dataset from 2013, we take only articles with more than 20 daily page views [Lat21]. This resulted in a collection of 463,819 knowledge source articles. We then proceed to turn these into knowledge source topics as described in Section 4.5. From this 463,819 knowledge source set, we proceed to determine the best 50,000 most likely knowledge source topics for our corpus. We experiment with various methods. The simplest, referred to as voting, entails ranking the knowledge source topic by the number of words showing up in the corpus. We also consider multiplying (Multiply) the count of the word in the corpus by the count of word in the knowledge source, as well as the addition of those two (Add). Other techniques such as ranking and term frequency inverse document frequency (tf-idf) hit memory and execution walls and were not considered in isolation. However, from the 50,000 knowledge source topic set we were able to compare ranking and tf-idf techniques. From this 50,000 knowledge source topic set we then compare how well each algorithm does against what we know to be topics contained in the human curated knowledge source. For each technique we choose the best 10,000 topics and compare how many in the 50,000 topic set were discovered.

4.5.2.3.2 Experimental Results

From the results given in Figure 4.25, we see that tf-idf from the 50,000 vote set performs the best out of any two models combined. Also considered, but not shown here were basic models that attempted to discount the size of the knowledge source topic (such as dividing by the length of knowledge source). This is an intuitive approach since a knowledge source topic that contains a lot of words will naturally have a higher amount of words showing up in the corpus. We discover however that these results were not particularly good, so we omit them. We do however find a remarkably interesting discovery in this regard. While simple averages do not yield good results, ranking does quite well. We submit that there is an effective way to discount large knowledge source topics—but simple length division will not work. Ranking, on the other hand, seems to do a better job at discounting large knowledge source topics. This intuition is supported by the results as

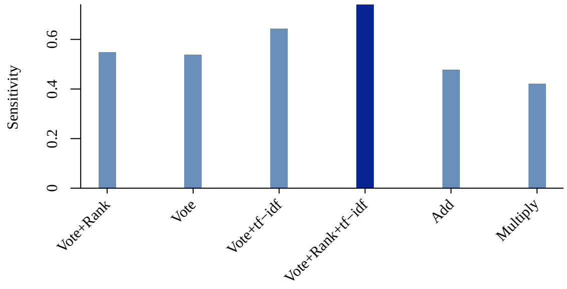


Figure 4.25: Sensitivity in finding knowledge source topics belonging to a corpus.

demonstrated by Figure 4.25. Quite surprisingly, there seem to be a subset of topics used in the corpus that ranking scores high while voting scores them in the middle. We propose a fascinating reason for this: large knowledge source topics need to be discounted for having lots of words but length divisions and tf-idf based scoring do not do as well as ranking. Ranking is able to associate "bad" knowledge source topics with other "bad" knowledge source topics by way of edges connected in a graph. Large knowledge source topics will tend to be connected more frequently to other large knowledge source topics so they serve to discount each other. This tendency works in opposition to voting or tf-idf, which tend to detect top scoring knowledge source topics as the most likely to be used in a corpus. So these two forces work in opposition placing the most likely topics to be in a corpus right at the middle. When we combine tf-idf, voting and raking we obtain the best results by far. This combination yields a significant result over the next best method (voting+tf-idf) at the 0.1 level.

4.5.2.4 Interpretability

To evaluate the effectiveness of our methodology we set up two human evaluated tasks to measure interpretability. For all experiments we use the datasets given in Table 4.20. The baseline methods used are: Infinite-LDA (InfTM) [Hei11], Hierarchical LDA (hLDA) [Ble03], the Nonparametric Topic Model (NTM) [Wal08], the Sawtooth Factorial Topic Embeddings Guided Gamma Belief Network (SawETM) [Dua21], the Nonparametric Tree-Structured Neural Topic Model (nTSNTM) [Che21a], and the Variationally-Learned Recurrent Neural Topic Model (VRTM) [Rez20]. VRTM was also defined to utilize outside information in the form of word embeddings [MSC13] and is evaluated as a separate model (VRTM+W2V). All baseline methods were parametrized according to their experiment descriptions in their respective papers. For the Interpretable Topic Model (IntTM) we use [Wal08] for Equation 4.101 and Equation 4.12 and Equation 4.16 for Equation 2.24 with their respective default parameters. To maximize interpretability we set $\xi = 1$ for IntTM. For weakly-supervised input we take the discovered knowledge source described in Section 4.5.1.3.

	Word Intrusion				**Topic Intrusion**			
	N	μ_1	MD	p-value	N	μ_1	MD	p-value
hLDA	600	0.15 ± 0.03	-0.02 ± 0.04	0.830	500	0.27 ± 0.04	0.02 ± 0.05	0.236
InfTM	600	0.15 ± 0.03	-0.01 ± 0.04	0.736	600	0.27 ± 0.04	0.02 ± 0.05	0.215
IntTM	600	$\mathbf{0.31 \pm 0.04}$	$\mathbf{0.14 \pm 0.05}$	**2.2e-09**	600	$\mathbf{0.36 \pm 0.04}$	$\mathbf{0.11 \pm 0.05}$	**1.3e-05**
NonTM	600	0.12 ± 0.03	-0.04 ± 0.04	0.987	600	0.26 ± 0.03	0.01 ± 0.05	0.421
nTSNTM	600	0.15 ± 0.03	-0.01 ± 0.04	0.709	500	0.28 ± 0.04	0.03 ± 0.05	0.175
SawETM	600	0.15 ± 0.03	-0.02 ± 0.04	0.808	600	0.28 ± 0.04	0.03 ± 0.05	0.107
VRTM	600	0.11 ± 0.03	-0.05 ± 0.04	0.996	600	0.28 ± 0.04	0.03 ± 0.05	0.163
VRTM+W2V	600	0.12 ± 0.03	-0.04 ± 0.04	0.984	600	0.24 ± 0.03	-0.01 ± 0.05	0.656

Table 4.21: The p-value, mean (μ_1), mean difference (MD) and associated 95% confidence intervals for each model aggregated the datasets for both the word intrusion and topic intrusion tasks.

4.5.2.5 Word Intrusion

In the word intrusion task [CBG09], we run each topic model against a dataset and sample an output ϕ_i. We take the 5 highest scoring words from ϕ_i as our "key" words. From the least scoring 5% of words of ϕ_i we take the word which is the highest scoring in ϕ_j where $j \neq i$ as the "intruder" word. We take this last step intentionally to allow for a more competitive "intruder." We repeat this process for a total of 20 samples across all datasets and models. Next, we shuffle the "intruder" and "key" words and create a form which asks a human evaluator to choose the "intuder" word. The exact directions submitted were: *Find the word that does not belong to the set of words.* The form was placed on Amazon Mechanical Turk[14] and each question was assigned 5 different "workers."

We aggregated the 100 answers for each dataset and computed a t-statistic against the null hypothesis of random selection. Additionally, we compute the associated 95% confidence intervals of both the hypothesis mean (μ_1) and mean difference (MD) between the hypothesis and the null (μ_0) means. Table 4.21 shows the computed values along with the associated p-value.

4.5.2.6 Topic Intrusion

The topic intrusion task [CBG09] is similar to the word intrusion task in that we give a set of "key" items mixed in with an intrusive item and ask the human evaluator to find the intrusive item. After

[14] https://www.mturk.com

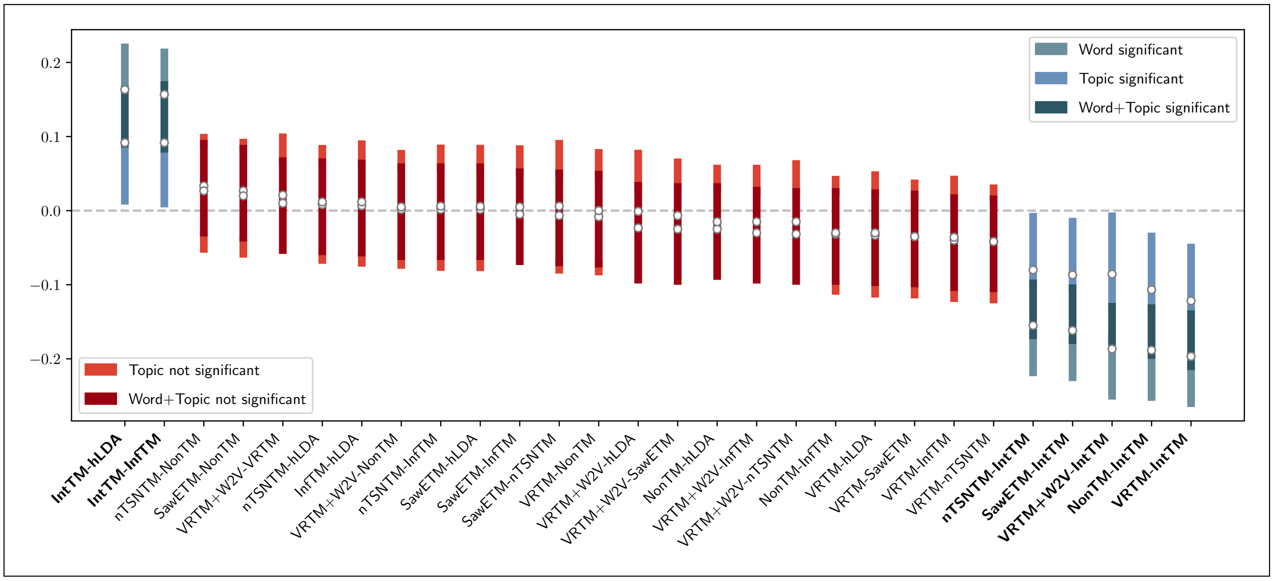


Figure 4.26: The Tukey-Kramer pairwise difference of means and associated 95% confidence intervals for the word and topic intrusion tasks.

topic modeling was complete for all models chose a random document d_i and the corresponding θ_i distribution. From θ_i, we take the highest 3 scoring topics as the "key" topics and from the the lowest scoring 5% topics we choose the topic which is the highest scoring in document d_j where $j \neq i$. The intuition behind this selection is the same as in Section 4.5.2.5. Each topic is represented by 8 of its highest scoring words and shuffled (only the topic order is shuffled, not the top words in the topic). We then create a form which presents the first 100 words of document d_i along with a selection to choose the "intruder" topic among the 3 total topics. The form also allows the user to click a button to see the full text of the document. We repeat the process for a total of 20 samples for each dataset. The form and samples are placed on Amazon Mechanical Turk and assigned to 5 workers each for a total of 100 questions per dataset. For the Wiki-20, dataset both hLDA and nTSNTM did not output enough topics to conduct the experiment, and were left off the evaluation for the Wiki-20 dataset.

After all questions were answered we compute the t-statistic and other statistical measures as we did in Section 4.5.2.5. The results are placed alongside the word intrusion topic in Table 4.21. Additionally, we seek to evaluate how well the models compare among themselves. Post-hoc analysis is conducted using the Tukey-Kramer method which represents the mean difference and 95% confidence intervals in Figure 4.26.

4.5.3 Discussion

In both the word intrusion and topic intrusion tasks, IntTM is the only model to achieve significance at the 0.01 level. In the word task, we see that all other models perform worse than the null hypothesis. We suspect this has to do with the experiment design. Among the "key" words to select from there may be a mixture of coherence along with more esoteric words. With a non consistent coherence, the human evaluator is not able to discern the overall topic and the intruder word becomes more favorable (of not being chosen as the intruder) than one or more of the esoteric words. One could argue that injecting outside information into the neural topic models could produce similar results to the IntTM. We do not deny this possibility, however we see that the addition of word embeddings does not significantly improve performance for VRTM. This suggests that more recent word embeddings, such as BERT [DCL19], may not necessarily lead to outperforming results to the IntTM.

Also of interest was the non significant difference between the Bayesian and neural topic models outside of IntTM. For the topic intrusion task, one could expect neural topic models to perform poorly since they tend to reuse individual θ distributions (see Section 4.5 for more details). However, that Bayesian models outside of IntTM perform similarly to neural topic models is a surprise. We hypothesis this similarity is due more to poor performance from the Bayesian models as opposed to good performance from the neural topic models. The non significance between Bayesian and neural models for the word task introduces an interesting area for investigation since the neural topic models produce topics with better perplexity and PMI-based scores. The inconsistency with perplexity and PMI-based metrics to our measure of interpretability is consistent with other interpretability studies [CBG09, DB21]. Our results may indeed add to the challenge of PMI-based methods being the most appropriate method for measuring interpretability [DB21]. However, we contend that PMI-based methods are still valid and useful. Especially since human evaluation is both costly and time-consuming.

This work investigates a novel combination of non-parametric Bayesian and weakly-supervised topic models. In this combination we discover a fascinating result—a self-contained non-parametric interpretable topic model. As we show with empirical results, this topic model can add interpretability into any non-parametric topic model, without the need to supply an existing knowledge source. This interpretability trade-off comes at the cost of perplexity—with the parameter ξ acting to determine how much interpretability to be added to the model. This novel method highlights a new approach to topic modeling—one in which topic interpretability is at the forefront of topic discovery.

CHAPTER 5

Conclusion

5.1 A summary of contributions

This work focuses on knowledge discovery from biomedical and scientific text. The first set of significant contributions involve our structured causal pipeline with the potential to enhance the scientific method. The initial step of our pipeline seeks to increase the amount of piecemeal causal text fragments. Next, we synthesize the causal assertions into a graphical form. From the graphical form, we develop a technique to quantify the strength of each assertion using Bayesian analytics. We then evaluate the causal network from a probabilistic framework which accounts for consistency in the equivalence class of derived constraints. In the final quantification step, we seek to apply graphical algorithms to add consideration to the contextual elements of piecemeal causal discovery. With the numeric considerations complete, we then proceed to increase our causal understanding of existing causal networks. We present to the biological research our findings which state which assertions are strongest, which elements are most likely to yield a causal connection, which elements in a community are most influential, and finally, which established biological causations are most likely due to confounding variables.

Starting from the raw text in scientific research papers, we provide a novel approach for deriving the causal elements. Our method underscores the integration of two distinct domains to produce causal discovery from raw text. These domains are bioinformatics-based sequence alignments and biological causality descriptions. Additionally, we expand on existing sequence alignment algorithms to provide an efficient and dynamic breakpoint sequence alignment. This technique supersedes existing dynamic alignment algorithms [AG11, HC03] and is well situated for contributions in the field of bioinformatics. In the domain of text-based biological causality, we find this technique to be well suited in conditions with a small amount of training data and a large amount of data for retrieval. The application of this method has uncovered causal chains which were previously unestablished.

With the casual textual elements obtained, we developed an approach for information synthesis, allowing for further knowledge discovery. Our causal elements take the form of a graph with connections denoting a causal relationship. The main contribution in this area is developiong a small finite set of causal semantics that represent a powerful range of biological relationships. These semantics are displayed as edge types between nodes representing biological entities. Connecting these causal fragments reveals a rich and powerful piecemeal causal network. An environment that is favorable for network-based analysis to enhance the scientific method.

To represent the strength between piecemeal causal assertions, we introduce a Bayesian-based scoring metric. We discover our method to be consistent with existing heuristic scoring methods within the biological community. These are convergency and consistency. The score assumes a generative model over a hidden distribution representing the true causal distribution of an element pair. We use an expectation-maximization to infer the posterior distribution over the set of classes. The expectation is normalized to finalize our score. The metric can easily be used to gauge the strength of evidence between pairs or used to analyze the key component of a study. When an entire research article is transcribed into the graphical form, the graphical network becomes a more succinct synthesis of the much larger amount of information contained in the research work.

From the piecemeal causal network, we move to constraint-based graphical deductions. Because each observation implies a causal connection, we can convert the implications into constraints representing statistical relations. These relations can then direct us to make decisions based on the inference of causal graphs consistent with the constraints. We discover the problem statement to be equivalent to a weighted boolean satisfiability problem (SAT). We submit our technique representing a novel synthesis of piecemeal causal network information as input into the state-of-the-art constraint-based answer set programming language Clingo [GKK11]. Starting from a piecemeal graphical network, we can construct the statistical relations to query Clingo about the set of possible causal graphs representing an equivalence class. We can then use the equivalence class to deduce what result is most consistent with the known information—an important step in hypothesis testing and experiment selection.

We then proceed to improve the introduced pipeline by considering the effect of contextual information. Considering the external connections between two biological elements leads us to develop an additional scoring criterion that weighs the effect of possible confounding variables. Similar to the PCEI, we institute a Bayesian model where the posterior probability represents the strength of evidence. The score is then combined with the PCEI to obtain a numeric indication of the strength of the evidence considering not only the biological elements under direct evaluation but also the surrounding variables. This scoring setting allows for a much more impactful constraint-based experiment selection and hypothesis testing. When used in an algorithm for equivalence class discovery, we show the technique to converge quicker than other techniques. This approach allows for scientific discovery faster and cheaper than before. We then show the applicability of this pipeline enhanced with contextual considerations over an exhaustive input set. We discover a set of biological elements that are likely to yield the most information gain when examined. We also discover biological pairs which are likely to be causally connected. In subset networks, we use our technique to determine which biological element is the most influential, a finding that can help targeted drugs and therapies isolate a biological element to obtain a maximal effect. We also posit biological elements in a network where the causal effect observed is likely due to confounding variables. This result can help scientists pinpoint which biological elements to avoid when targeting a specific entity or help draft experiments to discover confounding variables.

Another significant contribution this dissertation makes is in the topic modeling sub-domains of interpretability and labeling. Topic modeling has shown the ability to cluster tokens such that they resemble topics. A major drawback of existing methods is the tendency to assign semantically different terms together. The incohesive terms are acted upon solely based on being frequented often in the same set of documents. This deficiency is apparent in both Bayesian-based and neural topic models. To overcome this inadequacy, we develop a method that incorporates outside information into the topic model in the form of a knowledge source. Since the knowledge sources tend to be curated manually, they are assumed to be highly interpretable. It follows then that topic distributions biased from highly interpretable knowledge sources results in highly interpretable discovered topics. Indeed, we find this to be the case as human evaluators score our method higher in tests of interpretability. Since the knowledge sources are also labeled, we can easily transfer the label to the discovered topic. First, we introduce the foundation for this approach with the Source-LDA model. Next, we develop techniques to improve upon the pioneering model. We then relax more constraints leading to a complete, self-contained, highly interpretable non-parametric topic model with topic labels. The final work represents the state-of-art method for a simple and highly applicable topic model to discover labels and produce highly interpretable topics.

Bayesian topic models produce distributions over the vocabulary representing a topic and distributions over the topics themselves for each document in a corpus. The generative model, which forms the basis for inference, does not make any semantic determinations when assigning a word to a topic. It is in the assumption of topic modeling that words that frequently appear in the same documents belong to the same topic. This assumption can somewhat limit the interpretability of such models as semantically different words can show up frequently together, which results in incohesive topics. We contribute to the fields of interpretable topic modeling and topic labeling by introducing a novel concept of outside knowledge sources into the topic modeling generative model. We use the knowledge sources to form distributions assumed to be hyperparameters to the Dirichlet distributions that generate the topics. This method becomes more flexible than other existing methods leading to improved results. This balance results in higher interpretability, perplexity, and token accuracy than other models in the same field. When compared against both Bayesian-based and neural topic models, the interpretability is significantly higher. Additionally, this method serves to label the topics which make the model suitable for topic visualization and description applications.

Source-LDA provides the basis for interpretable topic modeling and topic labeling. However, there are a few shortcomings with the model. One such weakness is that the input is not always inclusive of every word that semantically belongs to that topic. We seek to mitigate this weakness by introducing context when deciding token assignments. In this aim, we capitalize on the recurrent neural network (RNN). We choose the RNN because the model is well-suited for context-based predictions Since each input of the RNN is trained on the complete history of all previous input, we can take advantage of this “memory” by asking the RNN whether a word belongs to a weakly supervised topic. By training our knowledge source on the RNN, we can determine the likelihood that a word belongs to a supervised input (knowledge source topic) and replace our naive weakly supervised predictive probability with the probability given by the RNN. This fusion further increases the perplexity of the model. We also show the benefits of this amalgamation in reverse. We demonstrate the ability of topic models to increase the predictive power of the RNN in our contrived model, Topic-RNN. The method resembles an ensemble approach of a set of independent RNNs trained over their respective topic partitions together with a vanilla RNN. Under our evaluation, Topic-RNN becomes the state-of-the-art RNN enhanced with topic modeling input.

We further improve weakly supervised topic models (including Source-LDA) by decreasing the running time taken to infer the topic clusters. The existing high running times of these models exist because the inference must consider the knowledge source input, which typically is large by design, and increases the iteration time by a multiplicative factor over the knowledge source size. The time increase limits the model to be run on a knowledge source input size in the order of 10^3. To increase the range of applicability of the model, we provide a pre-inference technique to rank the topics by how likely they are to be used in forming the output topics post-inference. Our method, KnowledgeRank, adapts PageRank to be used on graphical-based knowledge sources. We also demonstrate the ability to take text-based knowledge sources, form them into a graphical structure, and apply KnowledgeRank to rank text-based knowledge source input effectively. The same technique is also adapted to the weakly supervised model that improves perplexity and interpretability. The perplexity gains show the ability to compete, and, in some cases, outperform existing state-of-the-art neural topic models—a feat previously thought not to be likely.

KnowledgeRank effectively eliminates knowledge source input beforehand, increasing the range of input sets that can be used in weakly supervised models. However, KnowledgeRank is limited in the range of inputs. Although the input size is large, we present a technique to allow for an infinite input size. Our technique is to take advantage of the theoretical advantages conferred in the field of non-parametric topic models. On the path towards a nonparametric, infinite knowledge source input model, we contribute to the field of nonparametric topic modeling as well as Bayesian nonparametrics in general, in the development of our alternative view of the Dirichlet process, the biased coin flip process. Our novel view reimagines the Dirichlet process to be a partitioning of coins on a table into different bags. Each bag contains its own bias assumed for each coin toss in consideration of placing that coin in the bag. We prove this view to be equivalent to the Dirichlet process. The advantage of this alternative view comes from the reuse of existing topic model inference equations. Given the similarity of the view to the generative model of Bayesian topic modeling, we can easily transfer the topic modeling inference equations to the Dirichlet process. Since the topic modeling inference equations have entrenched their utility in topic discovery, it follows that similar inference equations may yield favorable results in non-parametric topic discovery. We indeed show this to be the case, as our model outperforms differing inference technique-based models. This work contributes not only to improved performance, but to the general understanding of the Dirichlet process as well.

Our final contribution is made by combining weakly supervised topic models with non-parametric topic models. Given the high interpretability of weakly supervised models and unbounded input size of nonparametric models, it is a naturally intuitive approach to achieve a superior model in the fusion of both models. We provide the theoretical framework and the model to realize this composition. We also provide a technique for constructing a knowledge source for any corpus without any prior information or input—an approach where we discover KnowledgeRank to be quite helpful. In the process of constructing our model, we realize the model becomes a complete self-contained nonparametric topic model for interpretability and topic labeling. Sans any prior input, outside of a corpus and hyperparameters, we can achieve topic discovery that is both highly interpretable and labeled. Compared against state-of-the-art neural topic models and established nonparametric Bayesian models, our method performs significantly better in human evaluated tasks of interpretability. We also find the unbounded input to mitigate the execution time barriers previously discussed in weakly-supervised topic models. We deliver to the research community a novel outperforming topic model that places interpretability and topic labeling at the forefront of topic discovery.

5.2 Key findings

This dissertation shows that sequence alignments of parts of speech text fragments can be used to extract causal assertions. Under the setting of a small amount of complex textual training data and a large amount of retrieval data, OpBerg outperforms machine learning-based approaches. OpBerg is most suitable for causality extraction among competing sequence alignment algorithms due to its flexibility. We can increase the number of causal assertions from a small amount of training data by transforming the training data into parts of speech text fragments and comparing the retrieval data using OpBerg. We also find that these matchings can easily extract agent and target pairs.

We also demonstrate that causal biological text can be effectively synthesized in graphical form. Using a small set of class labels, we can accurately describe the key components of a biological research article. When connected, this graph represents a valid network of piecemeal causality. We can then make interesting discoveries about the strength of connections, hypothesis testing and experiment planning from the connected elements.

The piecemeal cumulative evidence index is an effective measure of certainty of experimental validity. The score also quantifies the accumulation of qualitative evidence into a readily understood value. We find the value agrees with the biological concepts of consistency and convergency. This approach also serves to connect Bayesian statistics with biological heuristics. The score can help scientists focus on elements within a graphical network and easily convey meta-analytic information succinctly.

To test the strength of a hypothesis, we discover the piecemeal causal graph analysis to be beneficial. Analyzing the degrees of freedom for each element in the known piecemeal causal graph leads to discovering the true causal graph quicker and with less experimentation than a random-based algorithm. Also of benefit, is calculating the expectation of an effect alongside the degrees of freedom. An analysis that binds the possible set of graphs to statistical constraints can be used to determine an equivalence class which translates into an approach of true causal discovery. This analytic approach can determine the true nature of a causal network faster and more cost-effectively than performing experiments at random.

When assessing the meta-analytic observation of an experiment, this dissertation finds contextual information to play a notable role. In the process of equivalence class determination over a piecemeal causal graph, considering contextual information increases the convergence rate to the true causal graph. This consideration further improves the time and cost savings over random selection. We also show that a quantification of contextual information can be achieved similarly to the PCEI. Likewise, the scoring becomes consistent with consistency and convergency. In the evaluation of a less constrained and non-theoretical causal network, the contextual scoring further distinguishes important connections. This separation allows for a more focused view of scientific discoveries.

Our causal pipeline demonstrates that qualitative causal text fragments are sufficient to guide causal discovery. Over the exhaustive PubMed dataset, we make interesting discoveries about the causal world we have so far. These include: potential experiments that would maximize our understanding of the piecemeal causal world, which connections are the most important, experiments that are most likely to yield a causal relationship, and which variables are suspect to confounding variables. Our pipeline demonstrates that key parts of the scientific process can be automated and potentially increase the rate of discovery while driving costs down. We submit the theoretical basis alongside our findings and await validation from the biological domain.

To develop a topic model that labels topics and is highly interpretable, we show how to leverage outside knowledge sources to attach missing associations. Since both the Bayesian and neural topic model are lacking a prior understanding of semantic cohesiveness among words, incohesive words may be bound to the same topic at a high rate. To help partition the words to the proper topic we demonstrate our method, Source-LDA, to achieve higher interpretability and token accuracy than baseline models. The change required is only to the hyperparameters of the Dirichlet distribution. We find that the flexibility of a distribution drawn from Source-LDA's hyperparameters is a balance between a too rigid knowledge source influence and a completely free influence—resulting in better performance. The approach of biasing the hyperparameters represents a simple yet powerful mechanism that has the effect of making the topic assignments more cohesive.

To improve knowledge source topic models, this dissertation finds that considering context through the recurrent neural network aids topic assignments of words that do not exist in the vocabulary of the knowledge source. We demonstrate that there is a small similarity in the word construction per topic. We can use this slight difference to tilt the assignment of a word to a topic that is more like the words in the knowledge source article representing that topic. This technique results in better token assignment and perplexity. Additionally, we show how topic modeling inputs can increase word prediction of the RNN. When partitioned in an ensemble fashion per topic, the combination of a per topic RNN and an overall RNN result in an ensemble RNN that predicts both words and character better than any previous topic RNN hybrid model. We demonstrate that our combination asserts character-level RNNs as equally as predictive as word-level RNNs.

Another improvement to token assignment can be made using a combination of graphical-based and text-based knowledge sources. This process involves taking the text-based knowledge source and merging it into the graph-based knowledge source. With the merged graph, we can then run a weighted recursive ranking algorithm. Additionally, we can use a text-only knowledge source and apply the same technique of transformation and ranking to yield improved results. This ranking technique improves interpretability, perplexity, and token assignment. When used in the topic elimination phase during inference, ranking is superior to baseline methods. We also discover the ranking technique as an excellent approach to pre-inference knowledge source pruning. We find the relationship between knowledge source topic retrieval and rank filtered input size to be an inverse relationship. This ranking technique is demonstrated to be an effective method to allow knowledge source-based topic models a feasible execution time even with a very large knowledge source.

In the context of nonparametric topic models, we find an alternative view can help shape Dirichlet process inference. By taking a more topic-modeling interpretation of the Dirichlet process, we can quickly connect the interpretation to existing Dirichlet inference techniques found in topic modeling. This connection is key to constructing a non-parametric topic model with better perplexity and k-finding. Additionally, the approach represents a more tunable non-parametric topic model by supplying two scaling hyperparameters. When compared in Gaussian mixture models, the mixture approximations resemble kernel-based techniques.

In the combination of non-parametric and knowledge source topics, we find an improved interpretable topic model with topic labeling that is scalable to an infinite input size. In the theoretical basis for the generative model in this combination, we discover a parameter value can be used as input to specify the influence from the knowledge source input. This influence acts as an interpretability parameter with a range between 0 and 1, with 0 being low interpretability and 1 resulting in a higher perplexity. We find KnowledgeRank useful in filtering the knowledge source topics a priori to improve the combined topic model input. In our evaluation, this topic model significantly improves the interpretability of the discovered topics and the interpretability of the topics discovered in each document. We find that very little input is needed to discover and label highly interpretable topics, and we need not specify a knowledge source as input. This model demonstrates that we can have reasonable perplexity alongside interpretable, labeled topics discovered in a computationally efficient manner.

5.3 Future work

Although this dissertation contributes to knowledge discovery from biomedical and scientific text, much work is yet to be undertaken. In the context of piecemeal causal discovery, we highlight open research areas for each step in the pipeline. When discovering interpretable and labeled topics, we discuss current state-of-the-art approaches in general domains and their potential for improvement in our setting. Additionally, we explore application potentials in each sub-area.

5.3.1 The piecemeal causal pipeline

Of all the components in the piecemeal causal pipeline, the most raw piece is that of causal extraction. There exists a potential for many improvements in this area. Although it was shown that in the setting of low training data and a large retrieval space, OpBerg performs the best, it would be interesting to try and leverage existing knowledge bases and machine learning approaches to improve the results. We would assume an ensemble technique would be superior to OpBerg alone. OpBerg may be a good filtering technique however it still has problems with matching. A simple "not" in front of a biological element can drastically change the meaning of a text fragment, however OpBerg would only consider this as a single penalty (indel or mismatch). We may also be able to improve on weighing the matches. It may be the case that a match of adjectives should be less important than a match of nouns, however currently they are weighed the same. Execution time can be an area of improvement; one area in this space worth investigating is running the algorithm on the graphical processing unit (GPU). Any improvements would be particularly appealing for extremely large input sets. The last text extraction area we submit for future work is applying OpBerg to the domain of bioinformatics. In applications where existing dynamic algorithms are being used [AG11, HC03], OpBerg could be of use.

For the other components of the pipeline, we surmise that the graphical representation of ResearchMaps may be expanded to account for different types of assertions. We may consider an experiment conducted over three or more variables and create some way of integrating this into our existing schema. Another edge type may be investigated as well. This edge would be a two-way edge, where an effect was shown upon both variables—one in which the evidence is consistent with a causal effect in both directions. Alternatives to the PCEI may be fruitful in consideration. One may take the view that the score represents a probability of a future observation. Under this context, the problem becomes that of prediction. And with enough training data, prediction seems best suited under machine learning and deep learning approaches. The formulation of these models is left as an open research area. We may also seek to weigh the textual observations. Currently, each extracted text fragment is observed as a single observation. Perhaps two observations with different p-values should be considered differently. Also, we may be able to leverage sentiment analysis to score an assertion more highly if the text fragment reveals a high level of certainty or enthusiasm.

In our approach of using SAT solvers to discover an equivalence class, we should seek improvements in the execution time. Given the combinatorial nature of our problem, we are severely limited in the subgraph size we can input into the procedure. Currently, this can only be feasible for a network length less than 5. To improve the input size, it may be possible to leverage GPUs to run the computations in parallel. Other techniques such as simplifying assumptions, pruning, and other efficiencies should be explored as well. Existing parallel approaches may be used in isolation or in combination to investigate an improvement in applicability [MML11a, MML11b, MML12a, MML12b, MML12a, HS18]. For experiment selection, we maintain our approaches at a heuristic level. Theoretical work can be explored to establish the computation limits of our problem setting. The theoretical groundwork may also lay the foundation for an outperforming algorithm. Given the probabilistic formulation of our experiment's selection setting, it may be possible to leverage graphical-based deep neural networks to discover the next best experiment to perform. It may also be worthwhile to seek out ensemble-based methods that combine our existing approaches with novel techniques.

The pipeline overall need not be isolated to that of biological experiments. In any area of causal discovery, we suppose the piecemeal causal pipeline to be applicable. We can perhaps apply the pipeline to free-text causal statements in a general platform. For example, that of social media, we may start with a small number of labeled causal statements, expand on it with OpBerg, and end with a discovery of new causal elements to investigate. This approach also seems well suited to that of the medical domain. We may be able to contribute greatly to the study of diseases by taking our approach to confounding variables. It would be interesting to investigate some disorders such as autism to try and find a set of variables most influential in the onset of the disorder. This scope could also be expanded to include genetic markers as to understand genotype and phenotype relationships.

5.3.2 Topic interpretability and labeling

In the domain of topic interpretability and labeling, a significant area of future work is in that of evaluation. The practice of determining what a "good" discovery of topics is, is not a well-defined area. Perplexity, for example, is a well-defined metric—however, it is questionable how well perplexity and interpretability correlate [CBG09]. As we have shown in this dissertation, neural topic models that output high perplexity topics perform poorly in interpretability tasks. The same critique applies to pointwise mutual information (PMI) metrics. Although there is one study linking high PMI with high interpretability, recent work [New10, DB21] along with this dissertation provides evidence that challenges this metric as causal to interpretability. A well-defined and unquestionable metric of topic interpretability that is cheaply able to be obtained is an important goal to strive for in future research. The current best method, giving the topic modeling output to human annotators to determine interpretability, is both well-defined and unquestionable—however, the results are expensive and time-consuming to obtain.

Source-LDA represents a good first step for topic modeling with labeling and interpretability—however, there can still be some improvements. The assumption of the generative model assumes a normal distribution over the mapping of hyperparameters. This assumption presents difficulties in the inference, which requires an approximation to infer the posterior. It may be more appropriate to assume a Dirichlet distribution over the hyperparameters allowing for a more direct inference calculation. Additionally, the model assumes a uniformity of interpretability over its article set. There may be some work in this area to determine which one of the human-annotated input represents a poor topic and is ultimately discarded. The model also assumes a general equivalence of token count for each document. In documents that have more tokens, the model will give these documents a higher importance and likelihood of assignment since the higher token counts will more heavily influence the approximate posterior equation. It may lead to better results by assuming a proportional weighting or other weighting techniques.

A key discovery in our analysis of Rank-LDA was that in generated data, Bayesian topic modeling outperforms state-of-the-art neural topic models. This analysis should also be applied to non-generated datasets to investigate if the effect carries over. The presumption here is that it will not, and assuming that to be the case, it indicates the inferior performance may be due to the generative model. Other models should be considered, perhaps with different distributions or a different form of the model algorithm. It may be fruitful to look at dynamic generative models. These may be generative models that are built depending on the data instead of pre-defined. Under a dynamic model fit to the data, it may be possible to outperform neural topic models in terms of perplexity.

From our approach to combine topics and recurrent neural networks, it follows that the same technique could yield improvements in both convolutional neural networks (CNNs) and transformers. At its heart, Topic-RNN is an ensemble technique, and the concepts of specialization should apply to any other deep learning model. Additionally, it would be interesting to look at the comparison of ReSource-LDA in datasets against neural topic models. Perhaps a combination of ReSource-LDA and Rank-LDA could lead to an increase in gains over state-of-the-art neural topic models. The two models are compatible, so a sampling from either generated distribution should only require an additional draw from a Bernoulli distribution. In the inference calculations, this hidden variable can be sampled independently of the z assignment.

The biased coin-flip process has been shown to work well when the hyperparameters are known to be the same as the corpus. However, when the hyperparameters are unknown a priori, the values are determined as input. It would be favorable to have the parameters determined by the data. This approach may follow existing techniques of hyperparameter discovery [Hei11, Wal08, TJB06]. Further improvements should also be investigated as existing methods are less than ideal. The biased coin-flip process is also unique in that there are two input parameters, so the hyperparameter estimation may be more complicated. Also, work can be done to see how well the biased coin-flip process fairs in conjunction with our interpretable topic model. Existing methods are using previously established methods, and given the BCP's improvements over the established methods, it would imply even better results in the interpretable topic model.

The primary focus of the topic modeling portion of this thesis is that of Bayesian topic models. Although the neural topic model seems to be fashionable as of late, we do not find the output to be highly interpretable. The scoring method behind neural topic models, pointwise mutual information, has been recently challenged in its efficacy. Indeed, in our analysis, we do not find the correlation between PMI and interpretability to be high. Working towards an interpretable neural topic model is a worthwhile goal. We would encourage techniques to look to outside information. Even though in Section 4.5.2.4 we show existing methods that incorporate vector mappings do not show an improved effect, there may be a way this is possible. Outside of Word2vec, one can try Glove or BERT word embeddings. We also hypothesize a technique where knowledge sources are added alongside the neural topic model topic matrices to capture interpretability. In a similar manner to our Bayesian topic model, we can assume the knowledge source to be topics, and convert them to a probability value. Then we can feed this probability value as input in conjunction with the edges emitted from the topic matrices. In this way, the influence can be exerted on the topic matrices in much the same way as weakly-supervised Bayesian topic modeling.

This thesis presents the state-of-the-art topic model with interpretable topics, interpretable document assignments, and topic labels. However, we can seek improvement by adding into the model ReSource-LDA, Rank-LDA, and the BCP. If we can discover the parameters from the data, then the topic model becomes an entirely self-contained model with no necessary input outside of a corpus. The output will then represent the best topic discovery useful to applications that visualize or present the topics themselves. One example of this is the patient-adaptive retrieval summarization engine (PARSE)[1]. Further work must be done to combine the complete self-contained interpretable topic model with patient meta-data—however, once these steps have been completed, the engine behind PARSE is all but complete. Investigating the effect of PARSE on the patient-clinician interaction is a logical next step. This work has the potential to improve the understanding of a

[1] NIH National Library of Medicine, R21 LM011937

patient history leading to better outcomes—which could lead to drastic improvements in medical decisions and the prevention of medical errors.

REFERENCES

[AB90] Adelchi Azzalini and Adrian W Bowman. "A look at some data on the Old Faithful geyser." *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, **39**(3):357–365, 1990.

[acm] "ACM Computing Classification System." https://www.acm.org/publications/class-2012.

[AEB10] Corey W Arnold, Suzie M El-Saden, Alex AT Bui, and Ricky Taira. "Clinical case-based retrieval using latent topic analysis." In *AMIA annual symposium proceedings*, volume 2010, p. 26. American Medical Informatics Association, 2010.

[AG11] Alexej Abyzov and Mark Gerstein. "AGE: defining breakpoints of genomic structural variants at single-nucleotide resolution, through optimal alignments with gap excision." *Bioinformatics*, **27**(5):595–603, 2011.

[AG18] Berna Altınel and Murat Can Ganiz. "Semantic text classification: A survey of past and recent advances." *Information Processing & Management*, **54**(6):1129–1153, 2018.

[Aga21] Rohit Agarwal. "Phrases based Document Classification from Semi Supervised Hierarchical LDA." In *2021 2nd International Conference on Computation, Automation and Knowledge Management (ICCAKM)*, pp. 332–337. IEEE, 2021.

[AHB20] Ayse Berna Altinel, Mustafa Abdullah Hakkoz, Emre Barkin Bozdag, and Murat Can Ganiz. "Identifying Topic-based Opinion Leaders in Social Networks by Content and User Information." *International Journal of Intelligent Systems and Applications in Engineering*, **8**(4):214–220, 2020.

[Alt90] Stephen F Altschul et al. "Basic local alignment search tool." *Journal of molecular biology*, **215**(3):403–410, 1990.

[AMC19] Henrique F de Arruda, Vanessa Q Marinho, Luciano da F Costa, and Diego R Amancio. "Paragraph-based representation of texts: A complex networks approach." *Information Processing & Management*, **56**(3):479–494, 2019.

[AOC16] Corey W. Arnold, Andrea Oh, Shawn Chen, and William Speier. "Evaluating topic model interpretability from a primary care physician perspective." *Computer Methods and Programs in Biomedicine*, **124**:67–75, 2016.

[AX08] Amr Ahmed and Eric P. Xing. "Dynamic Non-Parametric Mixture Models and the Recurrent Chinese Restaurant Process: with Applications to Evolutionary Clustering." In *Proceedings of the SIAM International Conference on Data Mining, SDM 2008, April 24-26, 2008, Atlanta, Georgia, USA*, pp. 219–230, 2008.

[Bar03] Chitta Baral. *Knowledge Representation, Reasoning and Declarative Problem Solving*. Cambridge university press, 2003.

[Bec14] Marcus W Beck. “Average dissertation and thesis length.” https://github.com/fawda123/diss_proc, 2014.

[Bel61] Richard Bellman et al. “Some numerical experiments using Newton’s method for nonlinear parabolic and elliptic boundary-value problems.” *Communications of the ACM*, **4**(4):187–191, 1961.

[BFP21] Sergio Bacallado, Stefano Favaro, Samuel Power, and Lorenzo Trippa. “Perfect Sampling of the Posterior in the Hierarchical Pitman–Yor Process.” *Bayesian Analysis*, **1**(1):1–25, 2021.

[BG21] Maria Berger and Elizabeth Goldstein. “Increasing Sentence-Level Comprehension Through Text Classification of Epistemic Functions.” pp. 139–150, 2021.

[BGL08] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. “Fast unfolding of communities in large networks.” *Journal of statistical mechanics: theory and experiment*, **2008**(10):P10008, 2008.

[BHM09] A. Biere, M. Heule, and H. van Maaren. *Handbook of Satisfiability*, volume 185. IOS Press, 2009.

[Bia21] Federico Bianchi et al. “Pre-training is a Hot Topic: Contextualized Document Embeddings Improve Topic Coherence.” In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pp. 759–766, Online, 2021. Association for Computational Linguistics.

[Bla08] Eduardo Blanco et al. “Causal Relation Extraction.” In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*, 2008.

[Ble90] Guy E Blelloch. “Prefix sums and their applications.” 1990.

[Ble03] David M. Blei et al. “Hierarchical Topic Models and the Nested Chinese Restaurant Process.” In *Advances in Neural Information Processing Systems 16 [Neural Information Processing Systems, NIPS 2003, December 8-13, 2003, Vancouver and Whistler, British Columbia, Canada]*, pp. 17–24, 2003.

[Ble12] David M. Blei. “Probabilistic topic models.” *Commun. ACM*, **55**(4):77–84, 2012.

[BLF11] Halil Bisgin, Zhichao Liu, Hong Fang, Xiaowei Xu, and Weida Tong. “Mining FDA drug labels using an unsupervised learning technique - topic modeling.” *BMC Bioinformatics*, **12**(S-10):S11, 2011.

[BM07] David M. Blei and Jon D. McAuliffe. “Supervised Topic Models.” In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, pp. 121–128, 2007.

[BNJ01] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. “Latent Dirichlet Allocation.” In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pp. 601–608, 2001.

[BNJ03] David M Blei, Andrew Y Ng, and Michael I Jordan. “Latent Dirichlet Allocation.” *the Journal of machine Learning research*, **3**:993–1022, 2003.

[BRD20] Marc Beck, Syed Tahseen Raza Rizvi, Andreas Dengel, and Sheraz Ahmed. “From Automatic Keyword Detection to Ontology-Based Topic Modeling.” In *International Workshop on Document Analysis Systems*, pp. 451–465. Springer, 2020.

[BS21] Ananth Balashankar and Lakshminarayanan Subramanian. “Learning Faithful Representations of Causal Graphs.” In *ACL/IJCNLP 2021, August 1-6, 2021*, pp. 839–850. Association for Computational Linguistics, 2021.

[Bui10] Quoc-Chinh Bui et al. “Extracting causal relations on HIV drug resistance from literature.” *BMC Bioinformatics*, **11**:101, 2010.

[Cao21] Pengfei Cao et al. “Knowledge-Enriched Event Causality Identification via Latent Structure Induction Networks.” In *ACL/IJCNLP 2021, Virtual Event, August 1-6, 2021*, pp. 4862–4872. Association for Computational Linguistics, 2021.

[CAS16] Denise J Cai, Daniel Aharoni, Tristan Shuman, Justin Shobe, Jeremy Biane, Weilin Song, Brandon Wei, Michael Veshkini, Mimi La-Vu, Jerry Lou, et al. “A shared neural ensemble links distinct contextual memories encoded close in time.” *Nature*, **534**(7605):115, 2016.

[CBG09] Jonathan D. Chang, Jordan L. Boyd-Graber, Sean Gerrish, Chong Wang, and David M. Blei. “Reading Tea Leaves: How Humans Interpret Topic Models.” In Yoshua Bengio, Dale Schuurmans, John D. Lafferty, Christopher K. I. Williams, and Aron Culotta, editors, *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada*, pp. 288–296. Curran Associates, Inc., 2009.

[CC04] Du-Seong Chang and Key-Sun Choi. “Causal Relation Extraction Using Cue Phrase and Lexical Pair Probabilities.” In *Natural Language Processing - IJCNLP 2004, First International Joint Conference, Hainan Island, China, March 22-24, 2004, Revised Selected Papers*, pp. 61–70, 2004.

[Che21a] Ziye Chen et al. “Tree-Structured Topic Modeling with Nonparametric Neural Variational Inference.” In *ACL/IJCNLP 2021*, pp. 2343–2353. Association for Computational Linguistics, 2021.

[Che21b] Zifeng Cheng et al. “A Unified Target-Oriented Sequence-to-Sequence Model for Emotion-Cause Pair Extraction.” volume 29, pp. 2779–2791, 2021.

[CJ88] Ronald Christensen and Wesley Johnson. "Modelling accelerated failure time with a Dirichlet process." *Biometrika*, **75**(4):693–704, 1988.

[CLP21] Federico Camerlenghi, Antonio Lijoi, and Igor Prünster. "Survival analysis via hierarchically dependent mixture hazards." *The Annals of Statistics*, **49**(2):863–884, 2021.

[COG06] Adele P Chen, Masuo Ohno, K Peter Giese, Ralf Kühn, Rachel L Chen, and Alcino J Silva. "Forebrain-specific knockout of B-raf kinase leads to deficits in hippocampal long-term potentiation, learning, and memory." *Journal of neuroscience research*, **83**(1):28–38, 2006.

[Coh15] Paul R Cohen. "DARPA's Big Mechanism program." *Physical biology*, **12**(4):045008, 2015.

[CS03] Rui M Costa and Alcino J Silva. "Mouse models of neurofibromatosis type I: bridging the GAP." *Trends in molecular medicine*, **9**(1):19–23, 2003.

[CW08] Ronan Collobert and Jason Weston. "A unified architecture for natural language processing: deep neural networks with multitask learning." In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, pp. 160–167, 2008.

[CY10] Yong Cheng and Long-Chuan Yu. "Galanin protects amyloid-β-induced neurotoxicity on primary cultured hippocampal neurons of rats." *journal of Alzheimer's Disease*, **20**(4):1143–1157, 2010.

[Dar11] William M Darling. "A theoretical and practical implementation tutorial on topic modeling and gibbs sampling." In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pp. 642–647, 2011.

[DB21] Caitlin Doogan and Wray L. Buntine. "Topic Model or Topic Twaddle? Re-evaluating Semantic Interpretability Measures." In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pp. 3824–3848. Association for Computational Linguistics, 2021.

[DCL19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, 2019.

[DDL90] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. "Indexing by Latent Semantic Analysis." *JASIS*, **41**(6):391–407, 1990.

[dee12] "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups." *IEEE Signal Process. Mag.*, **29**(6):82–97, 2012.

[DGW20] Qing Deng, Yang Gao, Chenyang Wang, and Hui Zhang. "Detecting information requirements for crisis communication from social media data: An interactive topic modeling approach." *International Journal of Disaster Risk Reduction*, **50**:101692, 2020.

[DMG20] Alex Diana, Eleni Matechou, Jim Griffin, Alison Johnston, et al. "A hierarchical dependent Dirichlet process prior for modelling bird migration patterns in the UK." *Annals of Applied Statistics*, **14**(1):473–493, 2020.

[Do11] Quang Do et al. "Minimally Supervised Event Causality Identification." In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 294–303, 2011.

[Dua21] Zhibin Duan et al. "Sawtooth Factorial Topic Embeddings Guided Gamma Belief Network." In Marina Meila and Tong Zhang, editors, *ICML 2021, 18-24 July 2021, Virtual Event*, Proceedings of Machine Learning Research. PMLR, 2021.

[DWG17] Adji B. Dieng, Chong Wang, Jianfeng Gao, and John W. Paisley. "TopicRNN: A Recurrent Neural Network with Long-Range Semantic Dependency." In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[Ebe08] Frederick Eberhardt. "Almost Optimal Intervention Sets for Causal Discovery." In *UAI 2008, Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, Helsinki, Finland, July 9-12, 2008*, pp. 161–168, 2008.

[Ebe09] Frederick Eberhardt. "Introduction to the Epistemology of Causation." *Philosophy Compass*, **4**(6):913–925, 2009.

[Ebe10] Frederick Eberhardt. "Causal Discovery as a Game." In Isabelle Guyon, Dominik Janzing, and Bernhard Schölkopf, editors, *Causality: Objectives and Assessment (NIPS 2008 Workshop), Whistler, Canada, December 12, 2008*, volume 6 of *JMLR Proceedings*, pp. 87–96. JMLR.org, 2010.

[Ebe13] Frederick Eberhardt. "Experimental indistinguishability of causal structures." *Philosophy of Science*, **80**(5):684–696, 2013.

[Ebe17] Frederick Eberhardt. "Introduction to the foundations of causal discovery." *I. J. Data Science and Analytics*, **3**(2):81–91, 2017.

[EGS05] Frederick Eberhardt, Clark Glymour, and Richard Scheines. “On the Number of Experiments Sufficient and in the Worst Case Necessary to Identify All Causal Relations Among N Variables.” In *UAI ’05, Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence, Edinburgh, Scotland, July 26-29, 2005*, pp. 178–184, 2005.

[EGS06] F. Eberhardt, C. Glymour, and R. Scheines. “N−1 Experiments Suffice to Determine the Causal Relations Among N Variables.” In Dawn E. Holmes and Lakhmi C. Jain, editors, *Innovations in Machine Learning: Theory and Applications*, volume 194. Springer-Verlag, 2006.

[EKS96] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise.” In Evangelos Simoudis, Jiawei Han, and Usama M. Fayyad, editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pp. 226–231. AAAI Press, 1996.

[Est96] Martin Ester et al. “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise.” In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pp. 226–231, 1996.

[EW95] Michael D Escobar and Mike West. “Bayesian density estimation and inference using mixtures.” *Journal of the american statistical association*, **90**(430):577–588, 1995.

[Fer73] Thomas S Ferguson. “A Bayesian analysis of some nonparametric problems.” *The annals of statistics*, pp. 209–230, 1973.

[FGM07] Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. “The Infinite Tree.” In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*, 2007.

[Fis21] Jannik Fischbach et al. “Fine-Grained Causality Extraction From Natural Language Requirements Using Recursive Neural Tensor Networks.” In *29th IEEE International Requirements Engineering Conference Workshops, 2021*, pp. 60–69. IEEE, 2021.

[FJ15] Paul W Frankland and Sheena A Josselyn. “Memory allocation.” *Neuropsychopharmacology*, **40**(1):243, 2015.

[Fri04] Nir Friedman. “Inferring cellular networks using probabilistic graphical models.” *Science*, **303**(5659):799–805, 2004.

[GAA08] Daniel Gardner, Huda Akil, Giorgio A. Ascoli, Douglas M. Bowden, William J. Bug, Duncan E. Donohue, David H. Goldberg, Bernice Grafstein, Jeffrey S. Grethe, Amarnath Gupta, Maryam Halavi, David N. Kennedy, Luis N. Marenco, Maryann E. Martone, Perry L. Miller, Hans-Michael Müller, Adrian Robert, Gordon M. Shepherd, Paul W. Sternberg, David C. Van Essen, and Robert W. Williams. "The Neuroscience Information Framework: A Data and Knowledge Environment for Neuroscience." *Neuroinformatics*, **6**(3):149–160, 2008.

[GDG15] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. "DRAW: A Recurrent Neural Network For Image Generation." In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pp. 1462–1471, 2015.

[GE21] Jiaying Gong and Hoda Eldardiry. "Zero-shot Relation Classification from Side Information." In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 576–585, 2021.

[Geo21] Tsvetanka Georgieva-Trifonova. "Modified Pointwise Mutual Information-Based Feature Selection for Text Classification." In *Proceedings of the Future Technologies Conference*, pp. 333–353. Springer, 2021.

[GG11] Thomas L. Griffiths and Zoubin Ghahramani. "The Indian Buffet Process: An Introduction and Review." *J. Mach. Learn. Res.*, **12**:1185–1224, 2011.

[Gho21] Abdallah Ghourabi. "A BERT-based system for multi-topic labeling of Arabic content." In *2021 12th International Conference on Information and Communication Systems (ICICS)*, pp. 486–489. IEEE, 2021.

[Gir10] Roxana Girju et al. "A knowledge-rich approach to identifying semantic relations between nominals." *Inf. Process. Manage.*, **46**(5):589–610, 2010.

[GKK11] Martin Gebser, Benjamin Kaufmann, Roland Kaminski, Max Ostrowski, Torsten Schaub, and Marius Schneider. "Potassco: The Potsdam Answer Set Solving Collection." *AI Commun.*, **24**(2):107–124, 2011.

[GKM21] Greeshma N Gopal, Binsu C Kovoor, and U Mini. "Keyword Template Based Semi-supervised Topic Modelling in Tweets." In *International Conference on Innovative Computing and Communications*, pp. 659–666. Springer, 2021.

[GL88] M. Gelfond and V. Lifschitz. "The stable model semantics for logic programming." In *Logic Programming: Proceedings of the Fifth International Conference and Symposium*, pp. 1070–1080, 1988.

[Gly04] C. Glymour. "The automation of discovery." *Daedalus*, **133**(1):69–77, 2004.

[GM02] Roxana Girju and Dan I. Moldovan. "Text Mining for Causal Relations." In *Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference, May 14-16, 2002, Pensacola Beach, Florida, USA*, pp. 360–364, 2002.

[GMB99] John F Guzowski, Bruce L McNaughton, Carol A Barnes, and Paul F Worley. "Environment-specific expression of the immediate-early gene Arc in hippocampal neuronal ensembles." *Nature neuroscience*, **2**(12), 1999.

[Got82] Osamu Gotoh. “An improved algorithm for matching biological sequences.” *Journal of molecular biology*, **162**(3):705–708, 1982.

[Gra13] Alex Graves. “Generating sequences with recurrent neural networks.” *arXiv preprint arXiv:1308.0850*, 2013.

[Gri02] Tom Griffiths. “Gibbs sampling in the generative model of latent dirichlet allocation.” 2002.

[GS04] Thomas L Griffiths and Mark Steyvers. “Finding scientific topics.” *Proceedings of the National Academy of Sciences*, **101**(Suppl. 1):5228–5235, April 2004.

[Gu21] Yu Gu et al. “Domain-specific language model pretraining for biomedical natural language processing.” volume 3, pp. 1–23. ACM New York, NY, 2021.

[Gyo04] Zoltán Gyöngyi et al. “Combating Web Spam with TrustRank.” In *(e)Proceedings of the Thirtieth International Conference on Very Large Data Bases*, pp. 576–587, 2004.

[Han13] Joshua A. Hansen et al. “Probabilistic Explicit Topic Modeling Using Wikipedia.” In *Language Processing and Knowledge in the Web - 25th International Conference*, pp. 69–82, 2013.

[Han21] Rujun Han et al. “ESTER: A Machine Reading Comprehension Dataset for Reasoning about Event Semantic Relations.” In *EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pp. 7543–7559. Association for Computational Linguistics, 2021.

[Hav02] Taher H. Haveliwala. “Topic-sensitive PageRank.” In *Proceedings of the Eleventh International World Wide Web Conference*, pp. 517–526, 2002.

[HB12] Alain Hauser and Peter Bühlmann. “Characterization and greedy learning of interventional Markov equivalence classes of directed acyclic graphs.” *Journal of Machine Learning Research*, **13**:2409–2464, 2012.

[HC03] Xiaoqiu Huang and Kun-Mao Chao. “A generalized global alignment algorithm.” *Bioinformatics*, **19**(2):228–233, 2003.

[HC13] Neil Houlsby and Massimiliano Ciaramita. “Scalable Probabilistic Entity-Topic Modeling.” *CoRR*, **abs/1309.0337**, 2013.

[HEH13] Antti Hyttinen, Frederick Eberhardt, and Patrik O. Hoyer. “Experiment selection for causal discovery.” *J. Mach. Learn. Res.*, **14**(1):3041–3071, 2013.

[Hei08] Gregor Heinrich. “Parameter estimation for text analysis.” *University of Leipzig, Tech. Rep*, 2008.

[Hei11] Gregor Heinrich. “Infinite LDA implementing the HDP with minimum code complexity.” 2011.

[HEJ14] Antti Hyttinen, Frederick Eberhardt, and Matti Järvisalo. "Constraint-based Causal Discovery: Conflict Resolution with Answer Set Programming." In Nevin L. Zhang and Jin Tian, editors, *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence, UAI 2014, Quebec City, Quebec, Canada, July 23-27, 2014*, pp. 340–349. AUAI Press, 2014.

[HFH09] Mark A. Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. "The WEKA data mining software: an update." *SIGKDD Explorations*, **11**(1):10–18, 2009.

[HG08] Yang-Bo He and Zhi Geng. "Active learning of causal networks with intervention experiments and optimal designs." *Journal of Machine Learning Research*, **9**(Nov):2523–2547, 2008.

[HHE13] Antti Hyttinen, Patrik O. Hoyer, Frederick Eberhardt, and Matti Järvisalo. "Discovering Cyclic Causal Models with Latent Variables: A General SAT-Based Procedure." In Ann E. Nicholson and Padhraic Smyth, editors, *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, UAI 2013, Bellevue, WA, USA, August 11-15, 2013*. AUAI Press, 2013.

[HHK13] Ioana Hulpus, Conor Hayes, Marcel Karnstedt, and Derek Greene. "Unsupervised graph-based topic labelling using dbpedia." In Stefano Leonardi, Alessandro Panconesi, Paolo Ferragina, and Aristides Gionis, editors, *Sixth ACM International Conference on Web Search and Data Mining, WSDM 2013, Rome, Italy, February 4-8, 2013*, pp. 465–474. ACM, 2013.

[Hir75] Daniel S. Hirschberg. "A linear space algorithm for computing maximal common subsequences." *Communications of the ACM*, **18**(6):341–343, 1975.

[HKY07] Jin-Hee Han, Steven A Kushner, Adelaide P Yiu, Christy J Cole, Anna Matynia, Robert A Brown, Rachael L Neve, John F Guzowski, Alcino J Silva, and Sheena A Josselyn. "Neuronal competition and selection during memory formation." *science*, **316**(5823):457–460, 2007.

[HKY09] Jin-Hee Han, Steven A Kushner, Adelaide P Yiu, Hwa-Lin Liz Hsiang, Thorsten Buch, Ari Waisman, Bruno Bontempi, Rachael L Neve, Paul W Frankland, and Sheena A Josselyn. "Selective erasure of a fear memory." *Science*, **323**(5920):1492–1496, 2009.

[Hoa16] Cong Duy Vu Hoang et al. "Incorporating Side Information into Recurrent Neural Network Language Models." pp. 1250–1255, 2016.

[Hof99] Thomas Hofmann. "Probabilistic Latent Semantic Analysis." In Kathryn B. Laskey and Henri Prade, editors, *UAI '99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden, July 30 - August 1, 1999*, pp. 289–296. Morgan Kaufmann, 1999.

[HRS13] Joshua A. Hansen, Eric K. Ringger, and Kevin D. Seppi. “Probabilistic Explicit Topic Modeling Using Wikipedia.” In *Language Processing and Knowledge in the Web - 25th International Conference, GSCL 2013, Darmstadt, Germany, September 25-27, 2013. Proceedings*, pp. 69–82, 2013.

[HS18] Youssef Hamadi and Lakhdar Sais, editors. *Handbook of Parallel Constraint Reasoning*. Springer, 2018.

[Hua21] Wenti Huang et al. “Local-to-global GCN with knowledge-aware representation for distantly supervised relation extraction.” volume 234, p. 107565. Elsevier, 2021.

[Hus21] Musarrat Hussain et al. “A practical approach towards causality mining in clinical text using active transfer learning.” *Journal of Biomedical Informatics*, **123**:103932, 2021.

[HY10] Fei Huang and Alexander Yates. “Open-Domain Semantic Role Labeling by Modeling Word Spans.” In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pp. 968–978, 2010.

[IFD15] John P. A. Ioannidis, Daniele Fanelli, Debbie Drake Dunne, and Steven N. Goodman. “Meta-research: evaluation and improvement of research methods and practices.” *PLOS Biology*, **13**(10):e1002264, 2015.

[IJ02] Hemant Ishwaran and Lancelot F James. “Approximate Dirichlet process computing in finite normal mixtures: smoothing and prior information.” *Journal of Computational and Graphical statistics*, **11**(3):508–532, 2002.

[IJ03] Hemant Ishwaran and Lancelot F James. “Generalized weighted Chinese restaurant processes for species sampling mixture models.” *Statistica Sinica*, pp. 1211–1235, 2003.

[IS88] Alan J Izenman and Charles J Sommer. “Philatelic mixtures and multimodal densities.” *Journal of the American Statistical association*, **83**(404):941–953, 1988.

[JDR20] Lan Jiang, Ly Dinh, Rezvaneh Rezapour, and Jana Diesner. “Which Group Do You Belong To? Sentiment-Based PageRank to Measure Formal and Informal Influence of Nodes in Networks.” In *International Conference on Complex Networks and Their Applications*, pp. 623–636. Springer, 2020.

[JIU12] Jagadeesh Jagarlamudi, Hal Daumé III, and Raghavendra Udupa. “Incorporating Lexical Priors into Topic Models.” In Walter Daelemans, Mirella Lapata, and Lluís Màrquez, editors, *EACL 2012, 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, April 23-27, 2012*, pp. 204–213. The Association for Computer Linguistics, 2012.

[KAS] Manolis Kyriakakis, Ion Androutsopoulos, Artur Saudabayev, and Joan Ginés i Ametllé. “Transfer Learning for Causal Sentence Detection.” In *BioNLP@ACL 2019*, pp. 292–297.

[KB91] Randy M Kaplan and Genevieve Berry-Rogghe. “Knowledge-based acquisition of causal relationships in text.” *Knowledge Acquisition*, **3**(3):317–337, 1991.

[KDA20] Taiwo Kolajo, Olawande Daramola, Ayodele Adebiyi, and Aaditeshwar Seth. “A framework for pre-processing of social media feeds based on integrated local knowledge base.” *Information Processing & Management*, **57**(6):102348, 2020.

[KDF15] Dimitrios Kotzias, Misha Denil, Nando de Freitas, and Padhraic Smyth. “From Group to Individual Labels Using Deep Features.” In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 597–606, 2015.

[KEM05] Steven A Kushner, Ype Elgersma, Geoffrey G Murphy, Dick Jaarsma, Geeske M van Woerden, Mohammad Reza Hojjati, Yijun Cui, Janelle C LeBoutillier, Diano F Marrone, Esther S Choi, et al. “Modulation of presynaptic plasticity and learning by the H-ras/extracellular signal-regulated kinase/synapsin I signaling pathway.” *Journal of Neuroscience*, **25**(42):9721–9734, 2005.

[KF09] Daphne Koller and Nir Friedman. *Probabilistic Graphical Models - Principles and Techniques*. MIT Press, 2009.

[KF11] Erica Korb and Steven Finkbeiner. “Arc in synaptic plasticity: from gene to behavior.” *Trends in neurosciences*, **34**(11):591–598, 2011.

[KKK14] Jieun Kim, Jeong-Tae Kwon, Hyung-Su Kim, Sheena A Josselyn, and Jin-Hee Han. “Memory recall and modifications by activating neurons with elevated CREB.” *Nature neuroscience*, **17**(1):65–72, 2014.

[KMK10] Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. “SemEval-2010 Task 5 : Automatic Keyphrase Extraction from Scientific Articles.” In Katrin Erk and Carlo Strapparava, editors, *Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval@ACL 2010, Uppsala University, Uppsala, Sweden, July 15-16, 2010*, pp. 21–26. The Association for Computer Linguistics, 2010.

[KML12] Jeon-Hyung Kang, Jun Ma, and Yan Liu. “Transfer Topic Modeling with Ease and Scalability.” In *Proceedings of the Twelfth SIAM International Conference on Data Mining, Anaheim, California, USA, April 26-28, 2012.*, pp. 564–575, 2012.

[KP11] Haridimos Kondylakis and Dimitris Plexousakis. “Exelixis: evolving ontology-based data integration system.” In Timos K. Sellis, Renée J. Miller, Anastasios Kementsietsidis, and Yannis Velegrakis, editors, *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, Athens, Greece, June 12-16, 2011*, pp. 1283–1286. ACM, 2011.

[Kra17] Peter M. Krafft. *A Rational Choice Framework for Collective Behavior*. Ph.D. thesis, Massachusetts Institute of Technology, 2017.

[Kri12] Alex Krizhevsky et al. “ImageNet Classification with Deep Convolutional Neural Networks.” In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012.*, pp. 1106–1114, 2012.

[KRV20] Rico Krueger, Taha H Rashidi, and Akshay Vij. “A Dirichlet process mixture model of discrete choice: Comparisons and a case study on preferences for shared automated vehicles.” *Journal of choice modelling*, **36**:100229, 2020.

[KWJ04] R. D. King, K. E. Whelan, F. M. Jones, P. G. K. Reiser, C. H. Bryant, S. H. Muggleton, D. B. Kell, and S. G. Oliver. “Functional genomic hypothesis generation and experimentation by a robot scientist.” *Nature*, **427**(6971):247–252, 2004.

[Lat21] “The Unknown Perils of Mining Wikipedia.” https://blog.lateral.io/2015/06/the-unknown-perils-of-mining-wikipedia/, 2021.

[LBB17] Adam Lally, Sugato Bagchi, Michael Barborak, David W. Buchanan, Jennifer Chu-Carroll, David A. Ferrucci, Michael R. Glass, Aditya Kalyanpur, Erik T. Mueller, J. William Murdock, Siddharth Patwardhan, and John M. Prager. “WatsonPaths: Scenario-Based Question Answering and Inference over Unstructured Information.” *AI Mag.*, **38**(2):59–76, 2017.

[LC21] Youngseok Lee and Jungwon Cho. “Web document classification using topic modeling based document ranking.” *International Journal of Electrical & Computer Engineering (2088-8708)*, **11**(3), 2021.

[LDB08] Renaud Lambiotte, J-C Delvenne, and Mauricio Barahona. “Laplacian dynamics and multiscale modular structure in networks.” *arXiv preprint arXiv:0812.1770*, 2008.

[LDS21] Andy Lücking, Christine Driller, Manuel Stoeckel, Giuseppe Abrami, Adrian Pachzelt, and Alexander Mehler. “Multiple annotation for biodiversity: developing an annotation framework among biology, linguistics and text technology.” *Language Resources and Evaluation*, pp. 1–49, 2021.

[LGN11] Jey Han Lau, Karl Grieser, David Newman, and Timothy Baldwin. “Automatic Labelling of Topic Models.” In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pp. 1536–1545. The Association for Computer Linguistics, 2011.

[LGY20] Muhammad Luthfi, Satoshi Goto, and Osamu Ytshi. “Analysis on the Usage of Topic Model with Background Knowledge inside Discussion Activity in Industrial Engineering Context.” In *2020 IEEE International Conference on Smart Internet* *of Things, SmartIoT 2020, Beijing, China, August 14-16, 2020*, pp. 15–22. IEEE, 2020.

[LHM09] N. Le Novere, M. Hucka, H. Mi, S. Moodie, F. Schreiber, A. Sorokin, E. Demir, K. Wegner, M. I. Aladjem, S. M. Wimalaratne, et al. “The systems biology graphical notation.” *Nature Biotechnology*, **27**(8):735–741, 2009.

[Li07] Wei Li et al. “Nonparametric Bayes Pachinko Allocation.” In *UAI 2007, Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence, Vancouver, BC, Canada, July 19-22, 2007*, pp. 243–250, 2007.

[Lin07] Yi-Ling Lin et al. “Syndecan-2 induces filopodia and dendritic spine formation via the neurofibromin–PKA–Ena/VASP pathway.” *The Journal of cell biology*, **177**(5):829–841, 2007.

[Liu16] Lin Liu et al. “An overview of topic modeling and its current applications in bioinformatics.” *SpringerPlus*, **5**(1):1608, 2016.

[Liu17] Lin Liu et al. “Predicting protein function via multi-label supervised topic model on gene ontology.” *Biotechnology & Biotechnological Equipment*, **31**(3):630–638, 2017.

[LLF20] Lucas Lehnert, Michael L Littman, and Michael J Frank. “Reward-predictive representations generalize across tasks in reinforcement learning.” *PLoS computational biology*, **16**(10):e1008317, 2020.

[LM06] Wei Li and Andrew McCallum. “Pachinko allocation: DAG-structured mixture models of topic correlations.” In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006)*, pp. 577–584, 2006.

[LML21] Zhentao Liang, Jin Mao, Kun Lu, Zhichao Ba, and Gang Li. “Combining deep neural network and bibliometric indicator for emerging research topic prediction.” *Information Processing & Management*, **58**(5):102611, 2021.

[LPW08] Antonio Lijoi, Igor Prünster, Stephen G Walker, et al. “Bayesian nonparametric estimators derived from conditional Gibbs structures.” *The Annals of Applied Probability*, **18**(4):1519–1547, 2008.

[LS13] Anthony Landreth and Alcino J Silva. “The need for research maps to navigate published work and inform experiment planning.” *Neuron*, **79**(3):411–415, 2013.

[LSJ08] Simon Lacoste-Julien, Fei Sha, and Michael I. Jordan. “DiscLDA: Discriminative Learning for Dimensionality Reduction and Classification.” In *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 8-11, 2008*, pp. 897–904, 2008.

[Luo19] Li-xia Luo. “Network text sentiment analysis method combining LDA text representation and GRU-CNN.” *Personal and Ubiquitous Computing*, **23**(3-4):405–412, 2019.

[LW02] Andy Liaw and Matthew Wiener. "Classification and regression by randomForest." *R news*, **2**(3):18–22, 2002.

[Lyu21] Qing Lyu et al. "Zero-shot Event Extraction via Transfer Learning: Challenges and Insights." In *ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pp. 322–332. Association for Computational Linguistics, 2021.

[Man14] Christopher D. Manning et al. "The Stanford CoreNLP Natural Language Processing Toolkit." In *Association for Computational Linguistics (ACL) System Demonstrations*, pp. 55–60, 2014.

[Man21] Narges Manouchehri et al. "Batch and online variational learning of hierarchical Dirichlet process mixtures of multivariate Beta distributions in medical applications." *Pattern Anal. Appl.*, **24**(4):1731–1744, 2021.

[MAO21] Ryo Masumura, Taichi Asami, Takanobu Oba, and Sumitaka Sakauchi. "Hierarchical Latent Words Language Models for Automatic Speech Recognition." *Journal of Information Processing*, **29**:360–369, 2021.

[Mar06] R. S. Margalit et al. "Electronic medical record use and physician-patient communication: an observational study of Israeli primary care encounters." *Patient Education and Counseling*, **1**:131–141, 2006.

[Mat17] Nicholas J. Matiasz et al. "Computer-Aided Experiment Planning toward Causal Discovery in Neuroscience." *Frontiers in Neuroinformatics*, **11**:12, 2017.

[May11] Conor Mayo-Wilson. "The problem of piecemeal induction." *Philosophy of Science*, **78**(5):864–874, 2011.

[May14] Conor Mayo-Wilson. "The limits of piecemeal causal inference." *The British Journal for the Philosophy of Science*, **65**(2):213–249, 2014.

[May19] Conor Mayo-Wilson. "Causal identifiability and piecemeal experimentation." *Synth.*, **196**(8):3029–3065, 2019.

[McA06] Jon D. McAuliffe et al. "Nonparametric empirical Bayes for the Dirichlet process mixture model." *Stat. Comput.*, **16**(1):5–14, 2006.

[MCN13] Yashar Mehdad, Giuseppe Carenini, Raymond T. Ng, and Shafiq R. Joty. "Towards Topic Labeling with Phrase Entailment and Aggregation." In Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff, editors, *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pp. 179–189. The Association for Computational Linguistics, 2013.

[MD18] D. Malinsky and D. Danks. "Causal discovery algorithms: A practical guide." *Philosophy Compass*, **13**(1), 2018.

[Mei06] Qiaozhu Mei et al. “A probabilistic approach to spatiotemporal theme pattern mining on weblogs.” In *Proceedings of the 15th international conference on World Wide Web, WWW 2006, Edinburgh, Scotland, UK, May 23-26, 2006*, pp. 533–542, 2006.

[mes] “Medical Subject Headings.” https://www.nlm.nih.gov/databases/download/mesh.html.

[MFW09] Olena Medelyan, Eibe Frank, and Ian H. Witten. “Human-competitive tagging using automatic keyphrase extraction.” In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1318–1327. ACL, 2009.

[Min03] Tom Minka. “Bayesian inference, entropy, and the multinomial distribution.” *Online tutorial*, 2003.

[MLM07] David M. Mimno, Wei Li, and Andrew McCallum. “Mixtures of hierarchical topics with Pachinko allocation.” In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007)*, pp. 633–640, 2007.

[MM88] Eugene W Myers and Webb Miller. “Optimal alignments in linear space.” *Computer applications in the biosciences: CABIOS*, **4**(1):11–17, 1988.

[MML05] S. Meganck, B. Manderick, and P. Leray. “A decision theoretic approach to learning Bayesian networks.” Technical report, Technical report, Vrije Universiteit Brussels, 2005.

[MML11a] Ruben Martins, Vasco Manquinho, and Inês Lynce. “Parallel search for Boolean optimization.” In *RCRA International Workshop on Experimental Evaluation of Algorithms for solving problems with combinatorial explosion*, volume 11, pp. 26–59, 2011.

[MML11b] Ruben Martins, Vasco M. Manquinho, and Inês Lynce. “Exploiting Cardinality Encodings in Parallel Maximum Satisfiability.” In *IEEE 23rd International Conference on Tools with Artificial Intelligence, ICTAI 2011, Boca Raton, FL, USA, November 7-9, 2011*, pp. 313–320. IEEE Computer Society, 2011.

[MML12a] Ruben Martins, Vasco Manquinho, and Inês Lynce. “Clause sharing in parallel maxsat.” In *International Conference on Learning and Intelligent Optimization*, pp. 455–460. Springer, 2012.

[MML12b] Ruben Martins, Vasco M. Manquinho, and Inês Lynce. “Parallel search for maximum satisfiability.” *AI Commun.*, **25**(2):75–95, 2012.

[MMZ12] Xianling Mao, Zhaoyan Ming, Zheng-Jun Zha, Tat-Seng Chua, Hongfei Yan, and Xiaoming Li. “Automatic labeling hierarchical topics.” In Xue-wen Chen, Guy Lebanon, Haixun Wang, and Mohammed J. Zaki, editors, *21st ACM International*

Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012, pp. 2383–2386. ACM, 2012.

[MRD06] Ruth Stashefsky Margalit, Debra Roter, Mary Ann Dunevant, Susan Larson, and Shmuel Reis. "Electronic medical record use and physician–patient communication: an observational study of Israeli primary care encounters." *Patient education and counseling*, **61**(1):134–141, 2006.

[MS21] Leacky Muchene and Wende Safari. "Two-stage topic modelling of scientific publications: A case study of University of Nairobi, Kenya." *Plos one*, **16**(1):e0243208, 2021.

[MSC13] Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. "Distributed Representations of Words and Phrases and their Compositionality." In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pp. 3111–3119, 2013.

[MSZ07] Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. "Automatic labeling of multinomial topic models." In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, California, USA, August 12-15, 2007*, pp. 490–499, 2007.

[MT04] Rada Mihalcea and Paul Tarau. "TextRank: Bringing Order into Text." In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25-26 July 2004, Barcelona, Spain*, pp. 404–411. ACL, 2004.

[Mur01] Kevin P Murphy. "Active learning of causal Bayes net structure." 2001.

[MWD18] Nicholas J. Matiasz, Justin Wood, Pranay Doshi, William Speier, Barry Beckemeyer, Wei Wang, William Hsu, and Alcino J. Silva. "ResearchMaps.org for integrating and planning research." *PLOS ONE*, **13**(5):1–25, 05 2018.

[MWM] Olena Medelyan, Ian H Witten, and David Milne. "Topic indexing with Wikipedia.".

[MWW17a] Nicholas J. Matiasz, Justin Wood, Wei Wang, Alcino J. Silva, and William Hsu. "Computer-Aided Experiment Planning toward Causal Discovery in Neuroscience." *Front. Neuroinform.*, **2017**, 2017.

[MWW17b] Nicholas J. Matiasz, Justin Wood, Wei Wang, Alcino J. Silva, and William Hsu. "Translating literature into causal graphs: Toward automated experiment selection." In Xiaohua Hu, Chi-Ren Shyu, Yana Bromberg, Jean Gao, Yang Gong, Dmitry Korkin, Illhoi Yoo, and Huiru Jane Zheng, editors, *2017 IEEE International*

Conference on Bioinformatics and Biomedicine, BIBM 2017, Kansas City, MO, USA, November 13-16, 2017, pp. 573–576. IEEE Computer Society, 2017.

[MWW21] Nicholas J. Matiasz, Justin Wood, Wei Wang, Alcino J. Silva, and William Hsu. "Experiment Selection in Meta-Analytic Piecemeal Causal Discovery." *IEEE Access*, **9**:97929–97941, 2021.

[MZ05] Qiaozhu Mei and ChengXiang Zhai. "Discovering evolutionary theme patterns from text: an exploration of temporal text mining." In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005*, pp. 198–207, 2005.

[MZ06] Qiaozhu Mei and ChengXiang Zhai. "A mixture model for contextual text mining." In *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, August 20-23, 2006*, pp. 649–655, 2006.

[MZ12] Tomas Mikolov and Geoffrey Zweig. "Context dependent recurrent neural network language model." In *2012 IEEE Spoken Language Technology Workshop (SLT), Miami, FL, USA, December 2-5, 2012*, pp. 234–239. IEEE, 2012.

[NAS07] David Newman, Arthur U. Asuncion, Padhraic Smyth, and Max Welling. "Distributed Inference for Latent Dirichlet Allocation." In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, pp. 1081–1088, 2007.

[ND18] Thuc Nguyen and Phuc Do. "CitationLDA++ an Extension of LDA for Discovering Topics in Document Network." In *Proceedings of the Ninth International Symposium on Information and Communication Technology*, pp. 31–37, 2018.

[New10] David Newman et al. "Automatic Evaluation of Topic Coherence." In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics*. The Association for Computational Linguistics, 2010.

[Ngu15] Dat Quoc Nguyen et al. "Improving Topic Models with Latent Feature Word Representations." *Trans. Assoc. Comput. Linguistics*, **3**:299–313, 2015.

[NIF] "Neuroscience Information Framework [Internet]." https://neuinfo.org/.

[Nin20] Xuefei Ning et al. "Nonparametric Topic Modeling with Neural Inference." *Neurocomputing*, **399**:296–306, 2020.

[Nis01] Darryl Nishimura. "BioCarta." *Biotech Software & Internet Report: The Computer Software Journal for Scient*, **2**(3):117–120, 2001.

[NND17] Ho Duy Tri Nguyen, Trac Thuc Nguyen, and Phuc Do. "Creating Prior-Knowledge of Source-LDA for Topic Discovery in Citation Network." In *International Conference on Computational Science and Technology*, pp. 443–453. Springer, 2017.

[NW70] Saul B Needleman and Christian D Wunsch. "A general method applicable to the search for similarities in the amino acid sequence of two proteins." *Journal of molecular biology*, **48**(3):443–453, 1970.

[oad] "Outline of Academic Disciplines." https://en.wikipedia.org/w/index.php?title=Outline_of_academic_disciplines.

[OEA20] Berke Oral, Erdem Emekligil, Seçil Arslan, and Gülsen Eryigit. "Information Extraction from Text Intensive and Visually Rich Banking Documents." *Inf. Process. Manag.*, **57**(6):102361, 2020.

[Pai10] John Paisley. "A simple proof of the stick-breaking construction of the dirichlet process." 2010.

[PB21] Jasabanta Patro and Sabyasachee Baruah. "A Simple Three-Step Approach for the Automatic Detection of Exaggerated Statements in Health Science News." pp. 3293–3305, 2021.

[PBM99] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. "The PageRank Citation Ranking: Bringing Order to the Web." Technical Report 1999-66, Stanford InfoLab, November 1999.

[PC09] John W. Paisley and Lawrence Carin. "Hidden Markov models with stick-breaking priors." *IEEE Trans. Signal Process.*, **57**(10):3905–3917, 2009.

[PD21] Phu Pham and Phuc Do. "The approach of using ontology as a pre-knowledge source for semi-supervised labelled topic model by applying text dependency graph." *International Journal of Business Intelligence and Data Mining*, **18**(4):488–523, 2021.

[Pea95] Judea Pearl. "Causal diagrams for empirical research." *Biometrika*, **82**(4):669–688, 1995.

[Pea09] Judea Pearl. *Causality*. Cambridge University Press, Cambridge, UK, second edition, 2009.

[Pec10] Pavel Pecina. "Lexical association measures and collocation extraction." *Lang. Resour. Evaluation*, **44**(1-2):137–158, 2010.

[Pen] Jeffrey Pennington et al. "Glove: Global Vectors for Word Representation." In *EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1532–1543. ACL.

[PHG86] Marc Postman, John Peter Huchra, and Margaret J Geller. "Probes of large-scale structure in the Corona Borealis region." *The Astronomical Journal*, **92**:1238–1247, 1986.

[phy] "Physics Subject Headings." https://physh.aps.org/.

[PL88] William R Pearson and David J Lipman. "Improved tools for biological sequence comparison." *Proceedings of the National Academy of Sciences*, **85**(8):2444–2448, 1988.

[PLS15] Vladimir O Pustylnyak, Pavel D Lisachev, and Mark B Shtark. "Expression of p53 target genes in the early phase of long-term potentiation in the rat hippocampal CA1 area." *Neural plasticity*, **2015**, 2015.

[PMM21] K. Rajendra Prasad, Moulana Mohammed, and Noorullah R. Mohammed. "Visual topic models for healthcare data clustering." *Evol. Intell.*, **14**(2):545–562, 2021.

[PNI08] Ian Porteous, David Newman, Alexander T. Ihler, Arthur U. Asuncion, Padhraic Smyth, and Max Welling. "Fast collapsed gibbs sampling for latent dirichlet allocation." In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*, pp. 569–577, 2008.

[Pop59] K. Popper. *The Logic of Scientific Discovery*. Basic Books, 1959.

[PR08] Omiros Papaspiliopoulos and Gareth O Roberts. "Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models." *Biometrika*, **95**(1):169–186, 2008.

[PS20] Nikhlesh Pathik and Pragya Shukla. "IN-LDA: An Extended Topic Model for Efficient Aspect Mining." In *Congress on Intelligent Systems*, pp. 359–370. Springer, 2020.

[RB10] Cristina Rabascio and Francesco Bertolini. "Blood-based biomarkers for the optimization of anti-angiogenic therapies." *Cancers*, **2**(2):1027–1039, 2010.

[re3] "Relationship and Entity Extraction Evaluation Dataset." https://github.com/dstl/re3d/.

[reu] "Reuters-21578, Distribution 1.0 [Internet]." https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection/.

[Rez20] Mehdi Rezaee et al. "A Discrete Variational Recurrent Topic Model without the Reparametrization Trick." In *Advances in Neural Information Processing Systems 33*, 2020.

[RGS04] Michal Rosen-Zvi, Thomas L. Griffiths, Mark Steyvers, and Padhraic Smyth. "The Author-Topic Model for Authors and Documents." In *UAI '04, Proceedings of the 20th Conference in Uncertainty in Artificial Intelligence, Banff, Canada, July 7-11, 2004*, pp. 487–494, 2004.

[RHN09] Daniel Ramage, David Leo Wright Hall, Ramesh Nallapati, and Christopher D. Manning. "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora." In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, EMNLP 2009, 6-7 August 2009, Singapore, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 248–256, 2009.

[RMD11] Daniel Ramage, Christopher D. Manning, and Susan T. Dumais. "Partially labeled topic models for interpretable text mining." In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, August 21-24, 2011*, pp. 457–465, 2011.

[Rob73] Robert W Robinson. "Counting labeled acyclic digraphs." *New directions in the theory of graphs*, pp. 239–273, 1973.

[Rog01] Torbjørn Rognes. "ParAlign: a parallel sequence alignment algorithm for rapid and sensitive database searches." *Nucleic acids research*, **29**(7):1647–1652, 2001.

[Ros61] Frank Rosenblatt. "Principles of neurodynamics. perceptrons and the theory of brain mechanisms." Technical report, CORNELL AERONAUTICAL LAB INC BUFFALO NY, 1961.

[RRH11] Thomas A. Russ, Cartic Ramakrishnan, Eduard H. Hovy, Mihail Bota, and Gully A. P. C. Burns. "Knowledge Engineering Tools for Reasoning with Scientific Observations and Interpretations: a Neural Connectivity Use Case." *BMC Bioinformatics*, **12**:351, 2011.

[RYM16] Asim J Rashid, Chen Yan, Valentina Mercaldo, Hwa-Lin Liz Hsiang, Sungmo Park, Christina J Cole, Antonietta De Cristofaro, Julia Yu, Charu Ramakrishnan, Soo Yeun Lee, et al. "Competition between engrams influences fear memory formation and recall." *Science*, **353**(6297):383–387, 2016.

[Sah21] Rupsa Saha et al. "Using Tsetlin Machine to discover interpretable rules in natural language processing applications." *Expert Systems*, p. e12873, 2021.

[Sai11] Tara Sainath et al. "Making deep belief networks effective for large vocabulary continuous speech recognition." In *2011 IEEE Workshop on Automatic Speech Recognition & Understanding*, pp. 30–35. IEEE, 2011.

[SCL14] Bingxing Shi, Sean D Conner, and Jian Liu. "Dysfunction of endocytic kinase AAK1 in ALS." *International journal of molecular sciences*, **15**(12):22918–22932, 2014.

[SFM97] Alcino J Silva, Paul W Frankland, Zachary Marowitz, Eugenia Friedman, George Lazlo, Dianna Cioffi, Tyler Jacks, and Roussoudan Bourtchuladze. "A mouse model for the learning and memory deficits associated with neurofibromatosis type I." *Nature genetics*, **15**(3):281–284, 1997.

[SG07] Mark Steyvers and Tom Griffiths. "Probabilistic topic models." *Handbook of latent semantic analysis*, **427**(7):424–440, 2007.

[SGP20] Dandan Song, Jingwen Gao, Jinhui Pang, Lejian Liao, and Lifei Qin. "Knowledge Base Enhanced Topic Modeling." In Enhong Chen and Grigoris Antoniou, editors, *2020 IEEE International Conference on Knowledge Graph, ICKG 2020, Online, August 9-11, 2020*, pp. 380–387. IEEE, 2020.

[SGS00] Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search, Second Edition*. Adaptive computation and machine learning. MIT Press, 2000.

[Shi10] Tadamichi Shimizu. “The role of macrophage migration inhibitory factor (MIF) in ultraviolet radiation-induced carcinogenesis.” *Cancers*, **2**(3):1555–1564, 2010.

[Shl20] Micah Shlain et al. “Syntactic Search by Example.” In *ACL 2020, Online, July 5-10, 2020*, pp. 17–23, 2020.

[SHM13] Emanuela Santini, Thu N Huynh, Andrew F MacAskill, Adam G Carter, Philippe Pierre, Davide Ruggero, Hanoch Kaphzan, and Eric Klann. “Exaggerated translation causes synaptic and behavioural aberrations associated with autism.” *Nature*, **493**(7432):411–415, 2013.

[SLB14] Alcino J. Silva, Anthony Landreth, and John Bickle. *Engineering the Next Revolution in Neuroscience: The New Science of Experiment Planning*. Oxford University Press, Oxford, 2014.

[SLN20] Yushu Shi, Purushottam Laud, and Joan Neuner. “A dependent Dirichlet process model for survival data with competing risks.” *Lifetime Data Analysis*, pp. 1–21, 2020.

[SM84] Gerard Salton and Michael McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, 1984.

[SM15] Alcino J Silva and Klaus-Robert Müller. “The need for novel informatics tools for integrating and planning research in molecular and cellular cognition.” *Learning and Memory*, **22**(9):494–498, 2015.

[SMH11] Ilya Sutskever, James Martens, and Geoffrey E. Hinton. “Generating Text with Recurrent Neural Networks.” In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pp. 1017–1024, 2011.

[SMY13] Derya Sargin, Valentina Mercaldo, Adelaide P Yiu, Gemma Higgs, Jin-Hee Han, Paul W Frankland, and Sheena A Josselyn. “CREB regulates spine density of lateral amygdala neurons: implications for memory allocation.” *Frontiers in behavioral neuroscience*, **7**, 2013.

[SOA16] William Speier, Michael K. Ong, and Corey W. Arnold. “Using phrases and document metadata to improve topic modeling of clinical reports.” *Journal of Biomedical Informatics*, **61**:260–266, 2016.

[SRR21] Ayan Sengupta, Suman Roy, and Gaurav Ranjan. “LJST: A Semi-supervised Joint Sentiment-Topic Model for Short Texts.” *SN Computer Science*, **2**(4):1–16, 2021.

[SSC11] Mark Steyvers, Padhraic Smyth, and Chaitanya Chemudugunta. “Combining Background Knowledge and Learned Topics.” *topiCS*, **3**(1):18–47, 2011.

[SSS20] Hadar Serviansky, Nimrod Segol, Jonathan Shlomi, Kyle Cranmer, Eilam Gross, Haggai Maron, and Yaron Lipman. "Set2graph: Learning graphs from sets." *Advances in Neural Information Processing Systems*, **33**, 2020.

[SSZ14] Yoshitake Sano, Justin L Shobe, Miou Zhou, Shan Huang, Tristan Shuman, Denise J Cai, Peyman Golshani, Masakazu Kamata, and Alcino J Silva. "CREB regulates memory allocation in the insular cortex." *Current Biology*, **24**(23):2833–2837, 2014.

[SW81] Temple F Smith and Michael S Waterman. "Identification of common molecular subsequences." *Journal of molecular biology*, **147**(1):195–197, 1981.

[SXW15] Xiangyan Sun, Yanghua Xiao, Haixun Wang, and Wei Wang. "On Conceptual Labeling of a Bag of Words." In Qiang Yang and Michael J. Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pp. 1326–1332. AAAI Press, 2015.

[SZ14] Cícero Nogueira dos Santos and Bianca Zadrozny. "Learning Character-level Representations for Part-of-Speech Tagging." In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pp. 1818–1826. JMLR.org, 2014.

[Tam17] Ardi Tampuu et al. "Multiagent cooperation and competition with deep reinforcement learning." *PloS one*, **12**(4):e0172395, 2017.

[Tau20] Hillel Taub-Tabib et al. "Interactive Extractive Search over Biomedical Corpora." In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing, BioNLP 2020, Online, July 9, 2020*, pp. 28–37, 2020.

[Teh06] Yee Whye Teh. "A Hierarchical Bayesian Language Model Based On Pitman-Yor Processes." In *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006*, 2006.

[TGG07] Yee Whye Teh, Dilan Görür, and Zoubin Ghahramani. "Stick-breaking Construction for the Indian Buffet Process." In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, AISTATS 2007, San Juan, Puerto Rico, March 21-24, 2007*, pp. 556–563, 2007.

[TJ97] Pasi Tapanainen and Timo Järvinen. "A non-projective dependency parser." In *5th Applied Natural Language Processing Conference, ANLP 1997, Marriott Hotel, Washington, USA, March 31 - April 3, 1997*, pp. 64–71, 1997.

[TJ07] Romain Thibaux and Michael I. Jordan. "Hierarchical Beta Processes and the Indian Buffet Process." In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, AISTATS 2007, San Juan, Puerto Rico, March 21-24, 2007*, pp. 564–571, 2007.

[TJB06] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. "Hierarchical dirichlet processes." *Journal of the american statistical association*, **101**(476):1566–1581, 2006.

[TKW07] Yee Whye Teh, Kenichi Kurihara, and Max Welling. "Collapsed Variational Inference for HDP." In *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, pp. 1481–1488, 2007.

[Tom20] Federico Tomasi et al. "Stochastic Variational Inference for Dynamic Correlated Topic Models." In Ryan P. Adams and Vibhav Gogate, editors, *UAI 2020, virtual online, August 3-6, 2020*, volume 124 of *Proceedings of Machine Learning Research*. AUAI Press, 2020.

[Top21] Topic. "Merriam-Webster.", 2021.

[TQL15] Duyu Tang, Bing Qin, and Ting Liu. "Document Modeling with Gated Recurrent Neural Network for Sentiment Classification." In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pp. 1422–1432, 2015.

[Vas10] Sophie Vasseur et al. "Hypoxia induced tumor metabolic switch contributes to pancreatic cancer aggressiveness." *Cancers*, **2**(4):2138–2152, 2010.

[Vat01] I. N. Vatcheva. *Computer-supported experiment selection for model discrimination*. Ph.D. thesis, University of Twente, Netherlands, 2001.

[Vel] Julien Velcin et al. "Readitopics: Make Your Topic Models Readable via Labeling and Browsing." In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018*.

[VJB06a] I. N. Vatcheva, H. de Jong, O. Bernard, and N. J. I. Mars. "Experiment selection for the discrimination of semi-quantitative models of dynamical systems." *Artificial Intelligence*, **170**(4–5):472–506, 2006.

[VJB06b] Ivayla Vatcheva, Hidde de Jong, Olivier Bernard, and Nicolaas J. I. Mars. "Experiment selection for the discrimination of semi-quantitative models of dynamical systems." *Artif. Intell.*, **170**(4-5):472–506, 2006.

[VJM00] Ivayla Vatcheva, Hidde de Jong, and Nicolaas J. I. Mars. "Selection of Perturbation Experiments for Model Discrimination." In Werner Horn, editor, *ECAI 2000, Proceedings of the 14th European Conference on Artificial Intelligence, Berlin, Germany, August 20-25, 2000*, pp. 191–198. IOS Press, 2000.

[Wad21] Takashi Wada et al. "Learning Contextualised Cross-lingual Word Embeddings and Alignments for Extremely Low-Resource Languages Using Parallel Corpora." In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pp. 16–31, 2021.

[Wal08] Hanna Megan Wallach. *Structured topic models for language*. Ph.D. thesis, University of Cambridge Cambridge, UK, 2008.

[Wan] Li et al Wan. “A Hybrid Neural Network-Latent Topic Model.”.

[Wan21] Zijian Wang et al. “Back to Prior Knowledge: Joint Event Causality Extraction via Convolutional Semantic Infusion.” In *PAKDD 2021, Virtual Event, May 11-14, 2021, Proceedings, Part I*, volume 12712 of *Lecture Notes in Computer Science*, pp. 346–357. Springer, 2021.

[War00] David J. Ward et al. “Dasher - a data entry interface using continuous gestures and language models.” In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology, UIST 2000, San Diego, California, USA, November 6-8, 2000*, pp. 129–137, 2000.

[WBS09] Yi Wang, Hongjie Bai, Matt Stanton, Wen-Yen Chen, and Edward Y. Chang. “PLDA: Parallel Latent Dirichlet Allocation for Large-Scale Applications.” In *Algorithmic Aspects in Information and Management, 5th International Conference, AAIM 2009, San Francisco, CA, USA, June 15-17, 2009. Proceedings*, pp. 301–314, 2009.

[Wel10] Ulrich Wellner et al. “ZEB1 in pancreatic cancer.” *Cancers*, **2**(3):1617–1628, 2010.

[Wer90] Paul J Werbos. “Backpropagation through time: what it does and how to do it.” *Proceedings of the IEEE*, pp. 1550–1560, 1990.

[Wik20] Wikipedia contributors. “Hirschberg’s algorithm — Wikipedia, The Free Encyclopedia.” https://en.wikipedia.org/w/index.php?title=Hirschberg%27s_algorithm&oldid=994689719, 2020. [Online; accessed 26-January-2021].

[Wik21] Wikipedia contributors. “Cancer — Wikipedia, The Free Encyclopedia.”, 2021. [Online; accessed 2-February-2021].

[WL07] Xueying Wang and Baojie Li. “Genetic studies of bone diseases: evidence for involvement of DNA damage response proteins in bone remodeling.” *International journal of biomedical science: IJBS*, **3**(4):217, 2007.

[WM06] Xuerui Wang and Andrew McCallum. “Topics over time: a non-Markov continuous-time model of topical trends.” In *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, August 20-23, 2006*, pp. 424–433, 2006.

[WMS09] Hanna M. Wallach, Iain Murray, Ruslan Salakhutdinov, and David M. Mimno. “Evaluation methods for topic models.” In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, pp. 1105–1112, 2009.

[WMS19] Justin Wood, Nicholas J. Matiasz, Alcino J. Silva, William Hsu, Alexej Abyzov, and Wei Wang. "OpBerg: Discovering causal sentences using optimal alignments.", 2019.

[WTW17] Justin Wood, Patrick Tan, Wei Wang, and Corey W. Arnold. "Source-LDA: Enhancing Probabilistic Topic Models Using Prior Knowledge Sources." In *33rd IEEE International Conference on Data Engineering, ICDE 2017, San Diego, CA, USA, April 19-22, 2017*, pp. 411–422, 2017.

[WWA21] Justin Wood, Wei Wang, and Corey Arnold. "The Biased Coin Flip Process for Nonparametric Topic Modeling." In *International Conference on Document Analysis and Recognition*, pp. 68–83. Springer, 2021.

[WWH10] Sinead Williamson, Chong Wang, Katherine A Heller, and David M Blei. "The IBP compound Dirichlet process and its application to focused topic modeling." In *ICML*, 2010.

[YHC20] Yi-Fan Yan, Sheng-Jun Huang, Shaoyi Chen, Meng Liao, and Jin Xu. "Active learning with query generation for cost-effective text classification." In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 6583–6590, 2020.

[YMY14] Adelaide P Yiu, Valentina Mercaldo, Chen Yan, Blake Richards, Asim J Rashid, Hwa-Lin Liz Hsiang, Jessica Pressey, Vivek Mahadevan, Matthew M Tran, Steven A Kushner, et al. "Neurons are recruited to a memory trace based on relative neuronal excitability immediately before training." *Neuron*, **83**(3):722–735, 2014.

[ZGX21] Zehua Zeng, Neng Gao, Cong Xue, Yuanye He, and Xiaobo Guo. "Learning from Audience Interaction: Multi-Instance Multi-Label Topic Model for Video Shots Annotating." In Weiming Shen, Jean-Paul A. Barthès, Junzhou Luo, Yanjun Shi, and Jinghui Zhang, editors, *24th IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD 2021, Dalian, China, May 5-7, 2021*, pp. 1075–1080. IEEE, 2021.

[Zha96] Tian Zhang et al. "BIRCH: An Efficient Data Clustering Method for Very Large Databases." In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, pp. 103–114, 1996.

[Zha21] Yue Zhang et al. "ReadsRE: Retrieval-Augmented Distantly Supervised Relation Extraction." In *SIGIR '21, Virtual Event, Canada, July 11-15, 2021*, pp. 2257–2262. ACM, 2021.

[ZWK09] Yu Zhou, Jaejoon Won, Mikael Guzman Karlsson, Miou Zhou, Thomas Rogerson, Jayaprakash Balaji, Rachael Neve, Panayiota Poirazi, and Alcino J Silva. "CREB regulates excitability and the allocation of memory to subsets of neurons in the amygdala." *Nature neuroscience*, **12**(11):1438–1443, 2009.

[ZZL15] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. "Character-level Convolutional Networks for Text Classification." In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 649–657, 2015.